Usability Heuristics for Fast Crime Data Anonymization in Resource-constrained Contexts

Aderonke Busayo, **SAKPERE** (olfade001@myuct.ac.za)

Submitted in fulfillment of the Requirements

for the degree of

Doctor of Philosophy

Faculty of Science

University of CapeTown

Friday 26th January, 2018

CERTIFICATION

As the candidate's supervisor, I have approved this thesis for submission.

Supervisor: Dr Anne V.D.M. Kayem

Signature: _____

Date: _____

DECLARATION

I declare that this thesis is my own work. Where collaboration with other people has taken place, or material generated by other researchers is included, the parties and/or materials are indicated in the acknowledgements or are explicitly acknowledged through references as appropriate.

This thesis is being submitted for the doctoral degree in Computer Science at the University of Cape Town. It has not been submitted to any other university for any other degree or examination.

Aderonke Busayo Sakpere

Date

ABSTRACT

This thesis considers the case of mobile crime-reporting systems that have emerged as an effective and efficient data collection method in low and middle income countries. Analyzing the data, can be helpful in addressing crime. Since law enforcement agencies in resource-constrained context typically do not have the expertise to handle these tasks, a cost-effective strategy is to outsource the data analytics tasks to third-party service providers. However, because of the sensitivity of the data, it is expedient to consider the issue of privacy. More specifically, this thesis considers the issue of finding low-intensive computational solutions to protecting the data even from an "honest-but-curious" service provider, while at the same time generating datasets that can be queried efficiently and reliably.

This thesis offers a three-pronged solution approach. Firstly, the creation of a mobile application to facilitate crime reporting in a usable, secure and privacy-preserving manner. The second step proposes a streaming data anonymization algorithm, which analyses reported data based on occurrence rate rather than at a preset time on a static repository. Finally, in the third step the concept of using privacy preferences in creating anonymized datasets was considered. By taking into account user preferences the efficiency of the anonymization process is improved upon, which is beneficial in enabling fast data anonymization.

Results from the prototype implementation and usability tests indicate that having a usable and covet crime-reporting application encourages users to declare crime occurrences. Anonymizing streaming data contributes to faster crime resolution times, and user privacy preferences are helpful in relaxing privacy constraints, which makes for more usable data from the querying perspective.

This research presents considerable evidence that the concept of a three-pronged solution to addressing the issue of anonymity during crime reporting in a resource-constrained environment is promising. This solution can further assist the law enforcement agencies to partner with third party in deriving useful

crime pattern knowledge without infringing on users' privacy. In the future, this research can be extended to more than one low-income or middle-income countries.

Keywords: Crime Reporting, Anonymization, Sliding Window Resizing, User Privacy, Resource-Constrained Environment.

Dedication

To my parents, who inspired me and provided a platform for my career dreams to come to fruition. Its sure not a coincidence having my PhD 35 years after my daddy's own.

To my late uncle, 'Remilekun and his family', the seed you have sown in my father's life birthed this. Its interesting having my PhD about 55 years after yours.

To my husband, who sacrificed a lot to support my research and quest for knowledge up to the finishing line.

ACKNOWLEDGEMENTS

I appreciate God Almighty who has made it possible for me to complete this research work successfully against all odds.

My sincere appreciation goes to my supervisor, Dr Anne V.D.M. Kayem, for exposing me to the rudiments of research. I lack words to express my gratitude for her constructive criticism at every phase of this research, her deep technical input, her creative writing style which I have adopted for this thesis, moral support and vast knowledge and values I have acquired as a result of my relationship with her. In truth, my life has been transformed as a result of my contact with her. Lastly, I thank her for the patience she exercised during the challenging period of this research. I am highly grateful to Prof James Gain for stepping in towards the end of my study when my supervisor left UCT. Prof, thank you for the sacrifices you made to ensure i finish strong. Besides my supervisors, I would like to thank Prof. Stephen Wolthusen from NTNU, Norway and the following UCT lecturers: Prof Hussein Suleman and Dr Melissa Densmore, for insightful comments. My sincere thanks also go to the Hasso Plattner Institute, University of Ibadan, IFIP, CROSSFYRE and INSTICC for the travel grant and scholarship given to me during my research.

I am grateful to all my colleagues in the infosec research group, ICT4D research laboratory and room 300. To my colleagues at the University of Ibadan (Nigeria), thank you for your support. In particular I would like to thank Dr Wunmi Isafiade, Dr Ayo Periola and Dr Olu Onifade for helping me proofread some chapters; Thabo Ndlovu who assisted in the design and implementation of the CryHelp App and Sadiq Hassan for his assistance during data analysis.

Last but not least, I would like to thank my family: parents, in-laws, siblings, uncles, aunts and pastors, for their unflinching support throughout my research and life in general, with a special mention of my husband, Wilson, for his sacrifices.

Contents

List of Figures

LIST OF ABBREVIATIONS

ABRS: Adaptive Buffer Re-Sizing Scheme

B-CASTLE: B-Continously Anonymizing Streaming Data via Adaptive Clustering

CM: Classification Metric

CASTLE: Continously Anonymizing Streaming Data via Adaptive Clustering

DGH: Domain Generalization Hierarchy

FAANST: Fast Anonymizing Algorithm for Numerical STreaming data

FADS: Fast Clustering-Based Anonymization for Data Streams

FAST: Fast Anonymization of Big Data Streams

IL: Information Loss

KIDS: K-Anonymization Data Stream Base on a Sliding Window

QI: Quasi-Identifier

SKY: Stream K-Anonymity

SR: Suppressed Records

SWAF: Sliding Window Anonymization Framework

Chapter 1

INTRODUCTION

This chapter presents a general introduction to the study. The context and motivation for the research are also presented. The motivation for this research is to propose usable and efficient methods of enabling data anonymization in resource-constrained contexts. Finally, the thesis outline is presented.

1.1 Overview

Information sharing is necessary to help policy makers in decision making [34]. This means that organizations and policy makers need to release data for mining purposes to guide effective decisions that will lead to achieving the goal of the organization. These data could be in various forms including personal information of customers and staff, and are usually quite sensitive. Thus, there is underlying fear that releasing such data to a third party could make it vulnerable to privacy violations.

In light of the aforementioned facts, organizations and individuals are usually reluctant to share personal information. This reluctance, as illustrated in Figure 1.1, poses a challenge to both organizations and third parties. However, the consequence of violating privacy is highly detrimental to the goals and objectives of organizations, since no effective policy can be formulated without mining data. Researchers are therefore constantly devising means to mitigate this challenge [85].

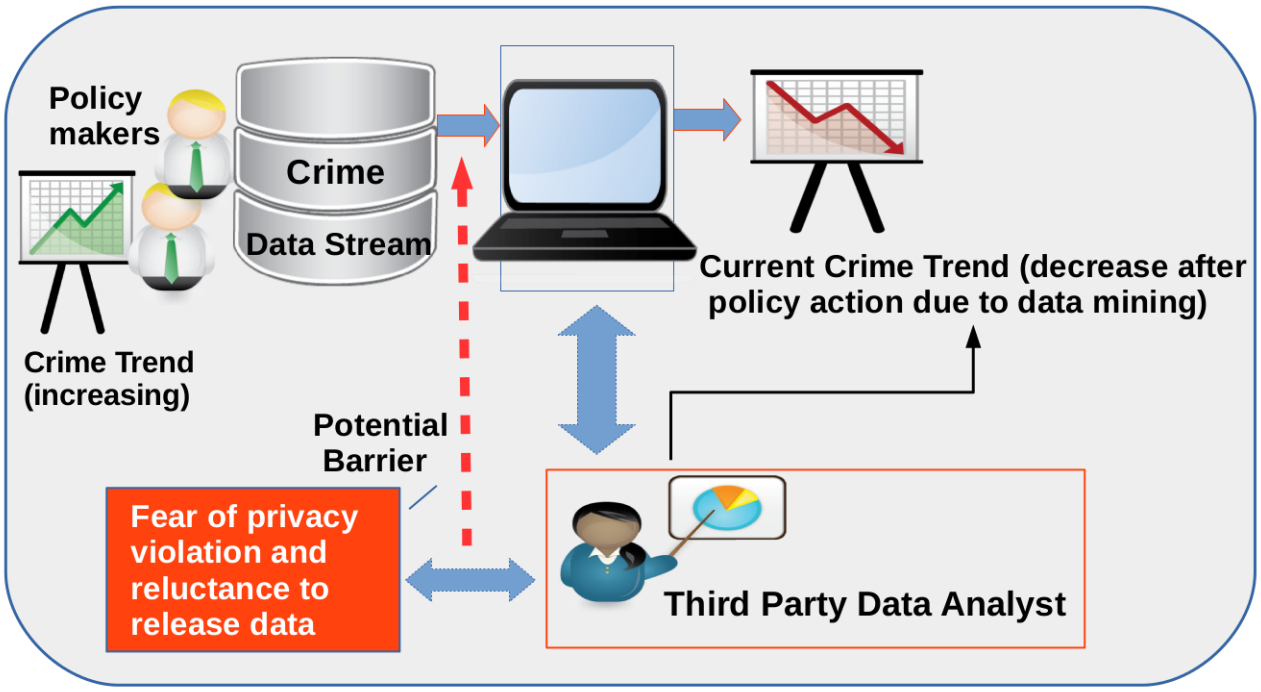


Figure 1.1: Depiction of Third-Party Challenges in Accessing Data

1.1.1 Context and Motivation

While organizations generate data that can contribute to improving performance daily, many of these organizations do not have the in-house expertise required to analyse the data. The lack of expertise is prominent in resource constrained environment (developing nations) where constraints on resources such as access to computational power, reliable electricity, and the Internet pose a further challenge. A cost-effective solution is to outsource the data to a professional third-party data analytics service provider. As illustrated in Figure 1.2, outsourcing often serves as a tug of war (barrier) between the organizations and the third party [68, 85].

While mining or analysing crime data in real-time is important to prevent or predict future crime occurrences, as depicted in Figure 1.1, a recent study [34] carried out in technologically resource-constrained environments has revealed that collected crime data are usually not studied or analysed to support crime resolution. A possible reason for this is the lack of the necessary in-house expertise, both in terms of human capital and computational processing power [40, 8, 73]. This deprives policy makers in these regions of the benefits that could have been derived through data analytics. A possible solution

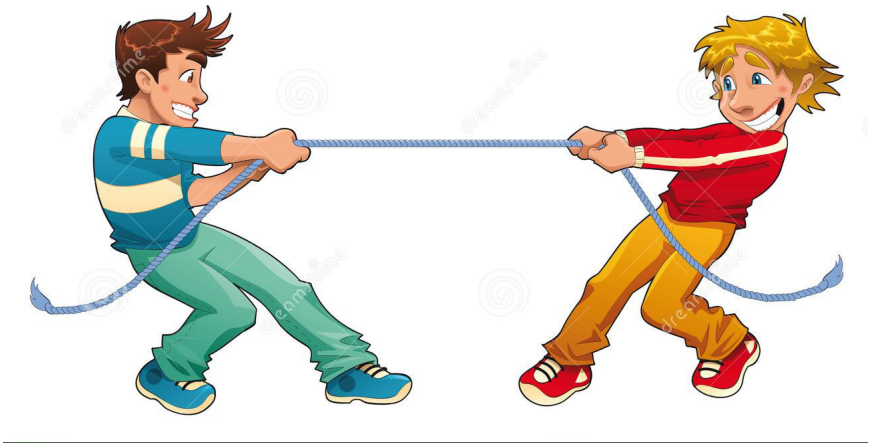


Figure 1.2: Tug of War Between Third Party Analyst and Organizations that Own Data (source: www.shutterstock.com)

to this is to involve a third-party data analytics service provider [34, 35, 68]. However, because of the sensitive nature of crime data it makes sense to ensure that the outsourced data are protected from all unauthorized access including that of an honest-but-curious data mining service provider.

To alleviate the challenge of protecting data released to a third party, a great number of privacy-preserving (anonymization) techniques, such as k-anonymity and other perturbation techniques, have been proposed [43, 85, 87]. However, these techniques are limited in terms of being usable in resource-constrained computing contexts. Therefore, this thesis is focused on developing a test bed framework to preserve privacy during information sharing of crime-reports in resource-constrained areas. This research considers the crime domain as the application domain for anonymization. However, it is important to stress that the ideas and approaches considered in this study are applicable to other areas as well, for example in a stock company that needs to anonymize its real-time sales data before releasing these to a third-party service provider to investigate its sales in order to adjust stock [71] or a hospital that needs to anonymize its daily medical records in real time or at intervals before releasing it to a third party for research purpose[46].

1.1.2 Trends in Data Anonymization

A naive approach to anonymizing the data before outsourcing to a third-party service provider is to remove explicit identifiers. Examples of explicit identifiers are the name, identity (ID) number, email

address and telephone number. It is then assumed that anonymity is maintained because the resulting data have been modified to exclude explicit identifiers. However, as Sweeney [83] pointed out, a major drawback of this naive approach to data anonymization is that sensitive details about a subject are still deducible through linking attacks which can reveal an individual's true identity. A linking attack can be provoked by combining quasi-identifiers, (QI), such as *date of birth, address* and *sex*, with external or publicly available tables or information. A QI is a set of non-explicit attributes that can sufficiently re-identify individual records when joined to external information that is publicly available and accessible to an adversary.

As an illustration of how a linking attack can be provoked, Figure 1.3, which shows two images is considered. The upper image contains a portion of a publicly available table in which *"name"* is an explicit identifier attribute and the lower image shows a portion of a data stream that has been sanitized to exclude explicit identifiers in order to disguise the identities of the individuals associated with the data. However, when a joining operation is performed on both images using the attributes common to both of them, the individual is re-identified successfully as Ronke who lives in 6 Alma Road, Rosebank and sensitive information about her is revealed.

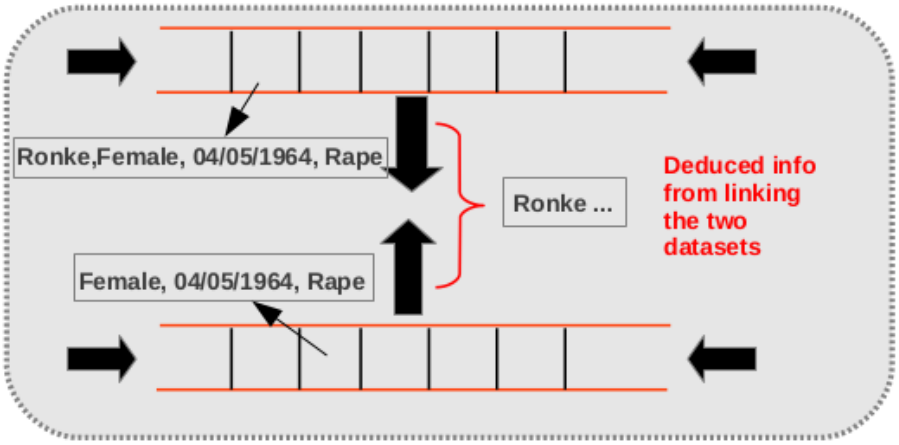


Figure 1.3: Crime Report Data Stream

Sweeney [83, 85] shows that 87% of the 1990 population in the United States were uniquely re-identified based only on three non-explicit identifier, namely a *five-digit ZIP, gender* and *date of birth* [82]. Therefore, simply removing explicit identifiers from a dataset does not guarantee data anonymity [83, 85] if other datasets with which a linking attack can be launched are available. Sweeney [85] proposed addressing this problem with an approach named k-anonymity to preserve the privacy of data.

K-anonymity achieves data preservation by hiding each individual in a set of at least *k* individuals in a way that an adversary might not get detailed individual information, but only information about a group of *k* individuals. In order to understand how k-anonymity works, assume an attacker tries to identify a friend in a k-anonymized table, but the only information he has is his birth date and gender. K-anonymity ensures that the adversary finds it difficult to identify the individual by guaranteeing that at least *k* people have the same date of birth and gender.The larger the *k*, the smaller the possible information gain of an adversary.

Samarati[74] and Sweeney[85] define k-anonymity as follows: Each released item of data must be such that every combination of values of QIs can be indistinctly matched to at least *k* individuals. K-anonymity uses two techniques, namely generalization and suppression [74, 84], to achieve anonymization. Generalization involves replacing (or recoding) a specific value with a general but semantically consistent value [84]. In order to demonstrate how generalization occurs, a specific year of birth such as '1984' can be generalized to '198*'; representing the year of birth in the interval '1980 - 1989', while suppression on the other hand withholds a value completely [84]. Suppression will replace a specific year of birth, '1984' with '****'.

K-anonymity is a privacy-preserving technique that ensures that a record in an anonymous table corresponds to at least *k* - 1 other records with respect to their QI attribute(s) where *k* is a pre-assigned integer variable and $k>1$ [74, 83, 85]. Table 1.1 shows data that need to be anonymized, while Table 1.2 is an anonymized data of Table 1.1 using k-anonymity, where $k = 2$ and QI = (date of birth, sex, address). According to Table 1.2, each sequence of values in QI has at least two occurrences. Hence, the probability of a linking attack occurrence is $1/k$.

Table 1.1: Crime Report Data Stream

Name	**Year of Birth**	**Sex**	**Address**	**Reported Crime**
Ronke	1989	Female	6 Alma Road, Rosebank	Rape
Wilson	1986	Female	10 Alma Road, Rosebank	Rape
Ayokunle Ola	1973	Male	10 Dikens Road, Salt River	Car Hijacking
Lydia Otoks	1975	Male	24 Dikens Road, Salt River	Burglary

In order to illustrate how k-anonymity works, assume that there is a need to protect the dataset in Table 1.1 against a linking attack. The first step is to remove any explicit identifier, which in this case is the name. The next step is to determine attributes that are QIs, namely date of birth and sex. Using the

generalization tree in Figure 1.4, some specific values on the QI attributes are replaced by some more general values. For example, a specific age value is replaced by an age range. Application of these steps leads to Table 1.2. That is, every record in the table belongs to a group of at least two tuples on the QI attributes. Therefore an attacker using sex and year of birth cannot re-identify any individual with confidence of more than *1/2*.

Table 1.2: Crime Report Data

Name	**Year of Birth**	**Sex**	**Address**	**Reported Crime**
*****	198*	Female	Rosebank	Rape
*****	198*	Female	Rosebank	Rape
*****	197*	Male	Salt River	Car Hijacking
*****	197*	Male	Salt River	Burglary

K-anonymity algorithms can generally be grouped into two categories, namely hierarchy-based generalization and hierarchy-free generalization [101]. In hierarchy-based generalization, the domain of each attribute is usually stated using a hierarchy (Domain Generalization Hierarchy: DGH) [38, 85]. The acceptable values of each attribute are usually constructed from the DGH, which results in Value Generalization Hierarchy (VGH) [101]. Figure 1.4 depicts the DGH and VGH of the attribute year of birth. The hierarchy-based generalization expects the data analyst to specify the DGH and VGH of each attribute before the generalization process begins [101]. The algorithms proposed by [38, 85] make use of hierarchy-based generalization. Hierarchy-free generalization uses clustering and partitioning to produce a generalized result. It does not require a user-defined generalization like hierarchy-based generalization. The algorithms proposed by [43, 87] are examples of hierarchy-free generalization that uses clustering.

Various research projects [52, 57, 80, 20] that have been carried out after the evolution of k-anonymity have led to the birth of newer privacy models to address the limits of k-anonymity. Some of the popular and newer privacy models that extend k-anonymity are ℓ-diversity and t-closeness.

ℓ-diversity attempts to address homogeneity attack to which k-anonymity is vulnerable by ensuring that each equivalence class in a k-anonymized table has at least ℓ distinct sensitive values. By equivalence class, it implies a set of records in an anonymized table that have the same values for the QIs.

Authors who developed t-closeness identified the limitation of ℓ-diversity in its assumption of adversarial knowledge. The authors argued that an adversary can gain information about a sensitive attribute as long as he/she has information about the global distribution of this attribute. Therefore, this deficiency

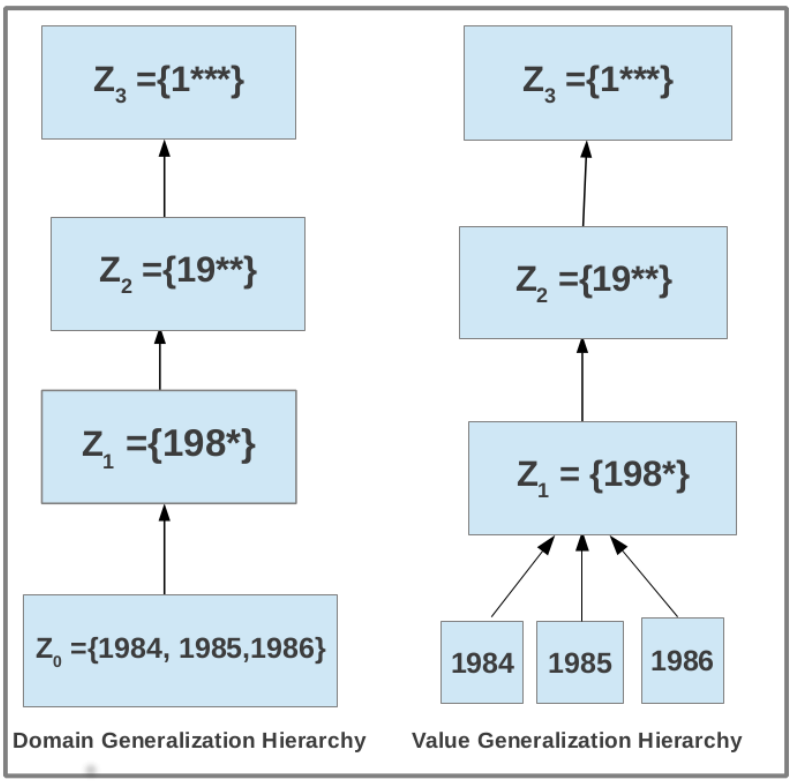


Figure 1.4: Generalization Hierarchies

was addressed by ensuring that the difference between the distribution of a sensitive attribute in any equivalence class and the distribution of the attribute is no more than a threshold t.

Another fast-growing privacy paradigm is differential privacy. Differential privacy achieves anonymization by altering the data (i.e. unanonymized data) with mathematical “noise". In other words, differential privacy preserves privacy through the “difference" between the data supplied and the noise added to it. Recent research [19, 79] have shown that t-closeness with k-anonymity can yield differential privacy.

The main motivation for embarking on this study is based on research [34] carried out in developing countries, which shows that crime data are usually collected without mining, studying or analysing. Mining or analysing crime data in real time is important to prevent or predict future crime occurrences. However, oftentimes the law enforcement agencies in developing countries are not equipped with the on-site expertise required to analyze the data efficiently in real time. It is therefore a cost-effective strategy [68] to transfer streaming crime data to a trusted third-party service provider for the anonymization process. Because of the sensitive nature of crime data it makes sense to ensure that the outsourced data is protected from all unauthorized access, including that of an “honest-but-curious” data mining

service provider.

1.2 Problem Statement

In the last decade, k-anonymity and its derivatives (l-diversity and t-closeness) have been preferred for the anonymization process for the following reasons:

1. It does not compromise the integrity (truthfulness) of anonymized data, [4, 85] making it useful for statistical purposes, research and data mining.
2. It produces anonymous data that meets legal and societal norms [38].

However, k-anonymity and its derivatives have been widely adopted for anonymizing static data efficiently, but are not directly suited to streaming data [10, 29, 50, 94, 101, 102]. To adapt k-anonymity and derivatives to data stream, a buffer (or sliding window) mechanism and delay constraint are introduced. The buffer is designed to hold a portion of the data stream and an anonymization algorithm is usually applied to the data in the buffer. Delay constraints ensure that each tuple does not stay in the buffer beyond its pre-defined deadline. However, many of the existing algorithms adapted for anonymization of data streams face the following challenges:

First, buffering according to delay constraints, can result in certain records being held in the buffer for long periods [101, 59, 71]. When such records are time-sensitive or need to be processed in real time, delay results in high levels of Information Loss (IL) from dropped records. As mentioned before, a key requirement of a good anonymization scheme is data utility. Therefore, high levels of IL due to expired tuples or dropped (or suppressed/unanonymizable) records are undesirable.

Second, building on the first problem, it was noted that many of the existing data stream anonymization schemes based on k-anonymity and its derivatives do not take distribution of future data streams into consideration during anonymization [29]. An implication of this is that a record that is likely to offer better anonymization at a lower rate of IL in a future sliding window or data stream can be anonymized with the current sliding window or data stream. Therefore, there is a need to have a model that can predict the best sliding window or stream with which a record should be anonymized.

Lastly, it is observed that existing anonymization schemes based on k-anonymity and its derivatives are structured to accept a static or constant privacy value for anonymization of an entire dataset [97, 72]. While this enhances data privacy, it has the drawback of not being practical for use in real-life situations. Typically, users have different privacy requirements, which should be captured in generating anonymized datasets.

1.3 Research Questions

1. Which of the privacy-preserving (anonymization) techniques is appropriate for anonymizing crime report data streams?

   To address this research question, first, a thorough literature review of existing anonymization schemes was conducted, highlighting their benefits and challenges. After the review, k-anonymity and its derivatives (ℓ diversity and t-closeness) were identified as the most appropriate. Then the adoption of k-anonymity and its derivatives on data stream was studied in order to come up with a framework of how these techniques can preserve privacy in a crime data stream.

2. How can a crime be reported in a secure and covert manner?

   To address this research question, a survey was carried out to identify the best platform that enables crime to be reported in a secured and covert manner. A mobile phone was identified as a platform that enables crime to be reported in a secured and covert manner. Therefore it was necessary to proceed to build a solution that enables crime to be reported securely using a mobile phone.

3. How can the anonymization scheme such as k-anonymity and its variants support data stream anonymization in a manner that reduces information loss and expired records?

   A tuple expires when it remains in the system for longer than a pre-specified threshold called delay [101, 59]. The term “delay" is a user-defined soft deadline that states for how long a tuple remain in the system. In order to minimize IL and expired tuples, the anonymization scheme was augmented with a time-based sliding window and Poisson probability distribution.

4. How can an anonymization process capture a user's privacy preference?

   A further survey was carried out to determine if the usage and integration of three-tier user-privacy into k-anonymity and derivatives are practicable in real life. Afterwards, the use of multinomial

regression and the association rule to automatically predict factors that influence users' privacy preference was conceived. As a further step, the results from the predictions of these birthed techniques were integrated into k-anonymity and its derivatives in order to aid determination of an appropriate privacy value to be used for the anonymization process for each individual.

1.4 Contributions

The contribution of this thesis is three-fold:

- The first step taken in this thesis was to create an application that allows people to report crime in a secured and covert way; as a result the crime reporting System in a university campus setting was digitized, precisely that of the University of Cape Town. For a successful implementation, the crime reporting solution was broken down into two components: front and back end. The system back end addresses the communication and storage of the application. The front end focuses on the development of the user interface.

- Next, it was necessary to adaptively resize the buffer of records in ways that minimize expired tuples, record suppression and resulting IL during anonymization. The proposed scheme, relies on the assumption that the record arrival rate obeys a Poisson distribution. The reason is that reported crime data follows Poisson property, which is a series of events occurring within a fixed time interval at an average rate that is independent of the time of the last event [53].

- Lastly, the concept of user privacy preferences in creating anonymized datasets was considered . There are two reasons for this. First, by taking into account user privacy preferences, anonymization of data can be boosted and consequently their transfer to these third-party data analytics service provider. Second, user requirements for privacy are considered in creating anonymized datasets, which is useful privacy preserving-wise. This was achieved with a three-tiered privacy scheme that works by using multinomial regression and the association rule to predict an appropriate privacy preference for a user. The idea of using a three-tiered privacy scheme hinges on results from past research that users find it easy to recognize privacy preferences if given three different privacy choices.

1.5 Publications

This thesis contains some ideas, figures and tables that have been published in the following recent articles:

Refereed Book Chapters

- Sakpere, A. B., and Kayem Anne V.D.M. "A State-of-the-Art Review of Data Stream Anonymization Schemes." Information Security in Diverse Computing Environments (2014): 24, **is solely based on chapter 2.**
- Sakpere, A. B., and Kayem Anne V.D.M. "Supporting Streaming Data Anonymization with Expressions of User Privacy Preferences." International Conference on Information Systems Security and Privacy (pp. 122-136). Springer International Publishing, **2015 is solely based on chapter 5.**

Refereed Conference Proceedings

- Sakpere, A. B., Kayem Anne V.D.M., and Ndlovu T. "A Usable and Secure Crime Reporting System for Technology Resource Constrained Context." In proceedings of 29th International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp 424 - 429, IEEE, 2015, Gwangju, North Korea, **is solely based on chapter 5.**
- Sakpere, A. B., and Kayem Anne V.D.M. "Adaptive Buffer Resizing for Efficient Anonymization of Streaming Data with Minimal Information Loss." In proceedings of 1st International Conference on Information Systems Security and Privacy (ICISSP), pp. 1-11, IEEE, 2015, ESEO, Angers, Loire Valley, France, , **is based on chapter 4.**
- Sakpere, A. B. "User-defined Privacy Preferences for k-Anonymization in Electronic Crime Reporting Systems for Developing Nations." In Proceedings of ICISSP Doctoral Consortium - DCISSP, pp. 13-18, 2015, ESEO, Angers, Loire Valley, France , **is solely based on chapter 5.**

Invited Talks/Refereed Poster Presentations

- Sakpere, A. B., and Kayem Anne V.D.M. "Dynamic Buffer Resizing for Efficient and Secure Streaming Data Anoymization". 9th Symposium on Future Trends in Service-Oriented Computing, Potsdam, Germany. June 25 - 27, 2014.
- Sakpere, A. B., and Kayem Anne V.D.M. "Effective Deadline Monitoring Framework for Data Stream Anonymization". Third International Workshop on Cryptography, Robustness, and Provably Secure Schemes for Female Young Researchers (CrossFyre), KU Leuven, June 20 - 21, 2013.
- Sakpere, A. B., and Kayem Anne V.D.M. "Has Data Stream Anonymization Reached its Full Potential?" Second International Invited Workshop on the Theories and Intricacies of Information Security problems (INTRICATE-Sec 2013) co-located with Information Security South Africa 13th Annual conference, August 14 -16, 2013, Johannesburg.

1.6 Outline

- Chapter 2: An extensive review of literature focusing on data stream anonymization is presented in this chapter. Section 2.3.1 and section 2.3.2 discuss perturbative and non-perturbative techniques. Detailed literature on k-anonymity, $\ell-$diversity and t-closeness is presented in section 2.4. Possible IL metrics are discussed in section 2.6, while data stream anonymization is presented in section 2.7.
- Chapter 3: This chapter presents the crime reporting framework to set the context for the data stream anonymization algorithms that is presented in Chapters 4 and 5 as well as the privacy analysis and experimental results in Chapter 6. The framework provides details on how crime reports are made and processed by the different modules which is elaborated on in Chapters 4 and 5. The framework is composed of modules such as 'Adaptive Buffering' and 'User Privacy Preferences'.
- Chapter 4: This chapter provides an extensive discussion on the first component of the crime report framework, which is the 'Adaptive Buffering' module. In this chapter a description supported by algorithmic schemas to show how the buffer size is adjusted to cope with the arrival rate without negatively impacting on IL is provided.

- Chapter 5: This chapter discusses how the user-privacy preference can be used to support the adaptive buffer resizing scheme in order to enhance privacy. The chapter begins by explaining the need to support k-anonymity with user privacy preferences. In order to determine the best user privacy preferences to integrate into the k-anonymity technique during data anonymization, it was necessary to conduct a real-life survey. Results from the real-life survey conducted indicate that a three-tiered privacy preference model is best suited for capturing user privacy expectations.

- Chapter 6: discusses the results of all the experiments. The experiments are centered on the usage of the CryHelp application (App) for crime reporting, the use of a buffering scheme modelled by the Poisson model and finally the integration of a three-tier user privacy preference.

- Finally Chapter 7: summarize the conclusion and main contributions of the thesis and provide some suggestions for future work.

1.7 Chapter Summary

This chapter began with an overview of the problem scenario that emerges in developing nations where the lack of data analytics expertise in law enforcement agencies makes it necessary for third-party data analytics provider intervene to aid in fast (crime) report analysis for knowledge support. In addition, it highlighted the fact that the growing need to make the processed information available to field officers requires a mechanism for capturing crime reports in real time and transferring these reports to the third-party service provider. While solutions in the literature that hinge on cryptography have been shown to be successful in protecting data in outsourced scenarios from unauthorized access, including that of “honest-but-curious" service providers, it is noted that querying encrypted streaming data is a time-consuming process and that the k-anonymization technique (as well as its derivatives such as: $\ell-$diversity and t-closeness) is a more practical approach to data privacy preservation in this case. However, the generic paradigm approach to privacy enforcement in these models needs to be refined in order to cater for individual needs. The focus in this thesis is on presenting a data-stream anonymization framework that addresses the limitations in existing framework. Therefore, this research emphasizes the need to integrate users' privacy preferences while attempting to reduce the delay caused by buffering.

Chapter 2

State of the Art

This chapter presents a general background to the study and provides a review of related research in this domain to reveal how previous research has attempted to resolve the issues of concern in privacy preservation. Furthermore, an extensive review of k-anonymity and its derivatives as a widely adopted privacy-preserving technique is documented, while emphasizing the shortcomings of these techniques in achieving effective data-stream anonymization. A recently emerging technique called “differential privacy" was discussed and the relationship between this technique and k-anonymity (and its variants) was pointed out. Furthermore, this chapter documents a brief introduction to the approach used to realize effective data-streaming anonymization in the previous research.

The rest of this chapter is divided into ten sections. The first section discusses the motivation for data privacy. The second focuses on the rationale and concept of data anonymization. The third explains the different techniques of data anonymization. The fourth focuses on three different techniques that can be used to enforce data privacy; these techniques are k-anonymity, ℓ-diversity and t-closeness. The fifth section focuses on new trends in privacy preservations, with particular emphasis on differential privacy, and also discusses the relationship between differential privacy and k-anonymity (and variants). The sixth section focuses on different metrics that measure the trade-off between data privacy and data utility. The seventh section focuses on privacy preservation in a data stream. The eight section focuses on principles that govern data stream anonymization. The ninth section focuses on different existing schemes and algorithms that can be used to achieve data stream anonymization. Finally, the tenth section summarizes the whole essence of this chapter and also re-iterates the contribution of this research work.

2.1 Motivation for Data Privacy

It is widely recognized that data need to be analyzed in order to extract useful information that can guide effective policies [34]. This need is known to be more pronounced and of very high necessity in certain domains, such as educational mining [62, 63] and crime mining [31, 36], to mention a few. Mining crime data is useful in several ways for achieving the following crime control targets:

- suspect prioritization
- hot-spot identification
- guiding patrol policies.

While these benefits are desirable and achievable, the lack of expertise needed to conduct data analysis in organizations compels them to outsource the (sensitive) data to a third party [68]. This compulsion presents the dilemma of outsourcing the data to a third party in order to enjoy the benefits of data analysis while simultaneously ensuring data privacy. Researchers have proposed several data anonymization approaches to address the challenge of maintaining data privacy before organisational data is released to a third party [83, 2]. In general, data anonymization approaches seek to encode sensitive information in the data in order to make the data less invasive. In what follows, an extensive review of data anonymization approaches is presented.

2.2 Data Anonymization Rationale and Concept

Data anonymization can simply be defined as a form of information sanitization aimed at protecting data privacy without impairing ease of analysis. It also entails the removal of personally identifiable information from datasets so that anonymity is maintained [83]. The purpose of data anonymization is to protect the privacy of individuals or end-users and to make it legal for governments and businesses to share their data. While there is a tendency to confuse access control (authentication) and data anonymization, it is important to recognize that data anonymization is different from access control and authentication [85], as presented in Table 2.1 .

Research into access control and authentication focuses on ensuring that the recipient of information has the authority or privilege to receive such information [85]. It focuses on safeguarding data access

Table 2.1: Difference between Data Anonymization and Access Control/Authentication

Data Anonymization	**Access Control/Authentication**
Focus on protection of users' privacy	Focus on users' authority and privileges
Safeguard against inference disclosures	Safeguard against direct disclosures

against direct disclosure and unauthorized access. Access control and authentication techniques do not guard against inferences that can be drawn from released data, which could be a breach of privacy [85]. Furthermore, cryptography, which is the encryption of data [58], will not be a suitable option for achieving data privacy because the authorized user (that is, the third-party analyst) will decrypt the data for analysis, therefore having access to the original data with no form of sanitization; this will not guard against inferences. This third-party analyst serves as a potential intruder who could breach data privacy at any point in the future. Furthermore, the study of the use of cryptographic techniques for protecting outsourced data from unauthorized access have shown to create a high overhead in terms of querying and updates, making analyzing large volumes of data in real-time a time-consuming process [44, 91]. Hence, data anonymization is considered a more viable solution to achieve data privacy. Therefore this chapter provides detailed literature on how information can be guarded against inferences that can be drawn from released data. In particular, the conventional and emerging techniques in data anonymization techniques as well as their usefulness and limitations are presented. Finally the research contributions in addressing some of these limitations are discussed.

2.3 Data Anonymization Techniques

In order to achieve data anonymization, one or more techniques are needed to make it impossible or at least more difficult for an intruder to identify a particular individual from stored data related to the person [2, 74, 14]. Anonymization techniques can generally be classified into two categories, namely:- perturbative and non-perturbative techniques[70]. Figure 2.1 presents the general classification of data anonymization, while Figure 2.2 presents a taxonomy of data anonymization techniques.

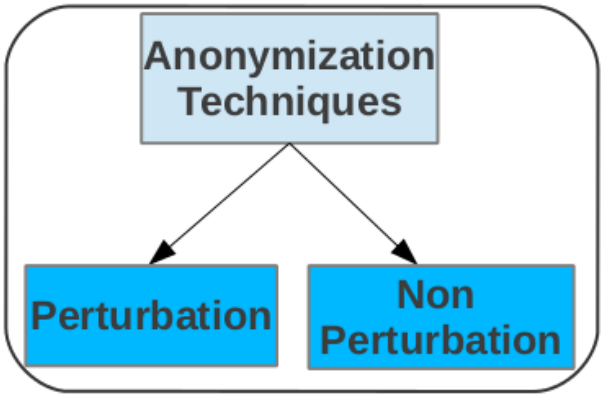


Figure 2.1: Depiction of the Classification of Anonymization Techniques

2.3.1 Perturbative Techniques

Perturbation techniques are techniques that attempt to mask confidential individual data elements while maintaining the underlying aggregate relationship with database. The words “perturbation" and “perturbative" are used interchangeably in this thesis.The use of perturbation entails introducing an external factor such as “noise" into the data, by modifying actual data values to conceal specific confidential information of individual record [95]. The purpose of the perturbation technique is to allow authorized users to access important aggregate statistics, such as averages and correlations, from the organization's database while protecting the individual identity of a record.

While the perturbation technique achieves the masking of individual confidential data elements, its use results in untruthful data. That is, it does not preserve the truthfulness of data [26, 42, 4]. As a result, the usefulness of such data in knowledge support is limited. To understand this better, let us assume the age of a crime victim is 30 years, using the perturbative technique, such a person's age may be changed to 60 in order to preserve privacy [70]. Examples of data anonymization techniques that fall under the perturbative technique are additive noise and swapping, among others.

1. **Additive Noise/Randomization:**

   This involves adding noise randomly by increasing or decreasing attribute values of individual records. This increase or decrease could be sufficiently large such that original values of individual records (i.e. the record's value before anonymization) cannot be re-identified [2]. Figure 2.3 shows an illustration of additive noise. From the figure, it is observed that each record is anonymized by offsetting it with a noise. Usually, this noise component, AN_i, is drawn from a probability

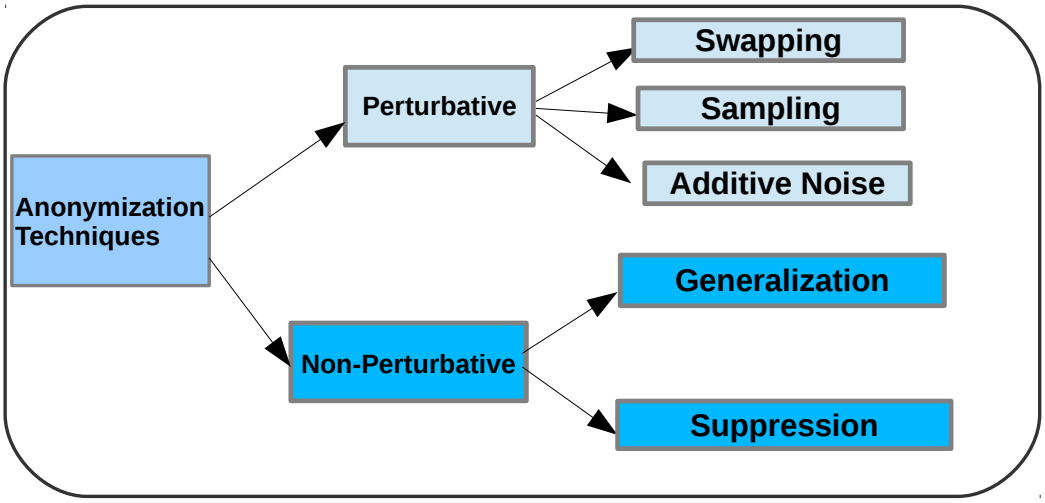


Figure 2.2: Depiction of Taxonomy of Data Anonymization Techniques

distribution and it is independent of the behavior of other records.

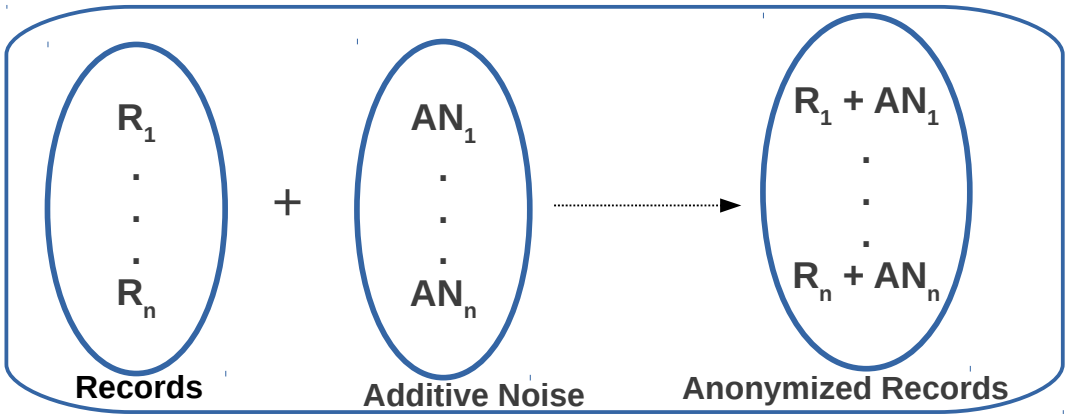


Figure 2.3: Depiction of Additive Noise

A major advantage of randomization method is its simplicity, and as a result it does not require a knowledge of the distribution of other records in the dataset. This implies that randomization can be implemented at the point of data collection and does not require a trusted server to store records before anonymization can take place. However, randomization is susceptible to adversarial attacks especially in cases of outliers and an attempt to further address this by adding more noise

will ultimately reduce the utility of data [2].

2. **Swapping:**

   Swapping preserves privacy by interchanging values associated with an attribute such that the value from the first row becomes that of the second row and vice versa [83]. As an illustration, let us assume Bob is a 20-year-old burglary victim while Charles is a 40-year-old theft victim. The use of swapping will change the age of Charles to that of Bob and vice versa. The use of such swapped data for crime analysis and mining with reference to age, as shown in Figure 2.4, is likely to provide a wrong intervention or solution.

   Worth noting is that this technique differs from randomization in that it does not allow the value of a record to be perturbed independently of the other records [2]. In other words, using data swapping, the perturbation (or anonymization) of a record depends on other records.

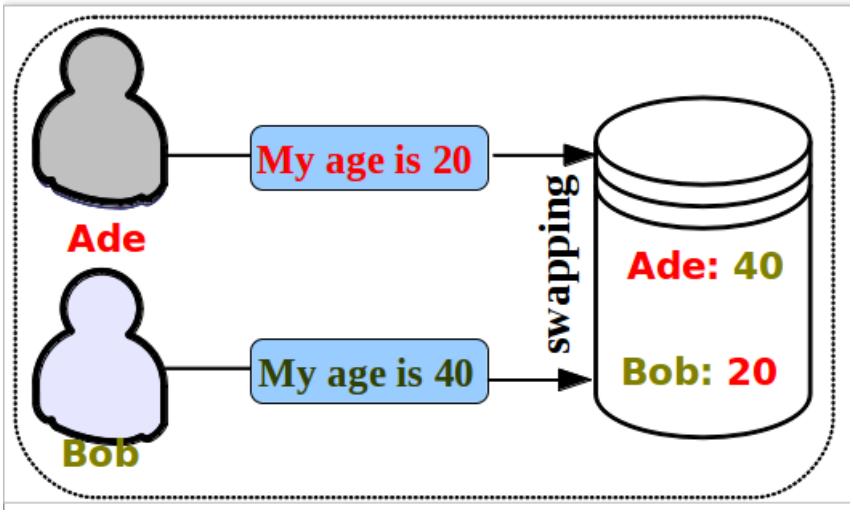


Figure 2.4: Depiction of swapping techniques

2.3.2 Non-Perturbative Techniques

Non-perturbation techniques achieve data anonymization or privacy by partially suppressing or reducing the detail of the original dataset [96, 18]. This means that no external "noise" is added to the records. As a result, non-perturbation techniques do not alter data. Thus, an important advantage of a non-perturbation technique is that it preserves data truthfulness [26, 42], because of its use of a generalization and a suppression mechanism.

1. **Generalization:**

   Generalization entails replacing a specific value with a more general value [74, 85]. Usually, possible generalization values are derived from a generalization hierarchy, where the root of the hierarchy has the most general values and the leaves correspond to most specific values. Therefore, generalization process typically occurs by replacing a specific value represented by the leaf nodes with a more general value represented by an ancestor node [14, 74]. Figure 2.5 shows a generalization hierarchy for an attribute crime type.

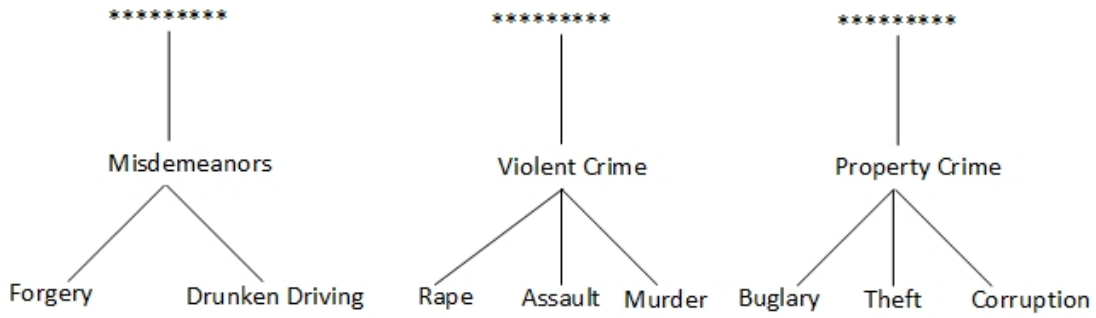


Figure 2.5: Generalization Hierarchy for the Attribute "Crime-Type"

2. **Suppression:**

   Suppression involves withholding a value completely, that is, the replacement of a specific value of an attribute or record with a missing or null value [15, 74, 85]. Suppression is usually used if a tuple cannot be anonymized or if the release of a tuple will significantly lead to data disclosure.

These techniques, that is, generalization and suppression, are widely used in k-anonymity, ℓ-diversity and t-closeness. More details about them follow in the next section, 2.4.

2.4 k-anonymity

A naive approach to achieve anonymization involves removing explicit information such as names and identification numbers that could serve as unique identifiers from publicly available databases and tables in order to ensure that private information is not leaked [8]. However, it is important to recognize that

using the aforementioned naive method does not guarantee the preservation of data privacy. Moreover, research has shown that the use of the naive approach is highly vulnerable to linking attack [85]. Linking attacks occur when sets of attributes (such as gender, birth-date) are used to link external data to uniquely identify a person in the database [57]. For example, Sweeney [85] capably discovered that after removing the explicit identifier of the US population, 87% of the people could be uniquely identified using harmless information such as birth date, gender and Zip codes. As a result, Sweeney developed the k-anonymity technique.

k-anonymity is a non-perturbative technique that makes use of generalization and suppression to achieve anonymization [74, 85, 84]. A first step for k-anonymity is to identify all attributes (i.e. QI) in the dataset that could be linked with an external dataset (examples of an external dataset include the census population of a particular country). Afterwards, k-anonymity preserves privacy by ensuring that each record of an anonymized (released) table corresponds to at least *k-1* other records with respect to their QI attribute, where k is a pre-assigned integer variable and $k > 1$ [74, 84, 85]. As an illustration of how k-anonymity works, assume an attacker attempts to identify a man, Walex, in the released table based on his birth-date, gender and Zip code, which the attacker knows, k-anonymity ensures there are $k - 1$ other individuals in the released table with the same birth date, gender and Zip code. As a result, k-anonymity is effective for counteracting linking attacks. In general, it reduces the probability of linking attacks to at least $1/k$ based on the generalization and suppression technique it adopts.

Generalization as earlier defined is simply a process of replacing (or recoding) a specific value with a more general but semantically consistent value [84]. For instance, a specific year of birth '1994' can be generalized to '199*', representing the year of birth in the interval '1990 to 1999'. On the other hand, suppression can be simply defined as the process of withholding a value completely [84, 85]. To illustrate suppression, a given specific year of birth, 1994, becomes ****. Following is an expository illustration of k-anonymity.

Table 2.2: Crime Report Data Stream

Row Number	**Name**	**Year of Birth**	**Sex**	**Address**	**Reported Crime**
1	Ronke	1989	Female	6 Alma Road, Rosebank	Rape
2	Wilson	1986	Female	10 Alma Road, Rosebank	Rape
3	Ayokunle Ola	1973	Male	10 Dikens Road,Salt River	Car Hijacking
4	Lydia Otoks	1975	Male	24 Dikens Road,Salt River	Burglary

Table 2.3: Anonymized Crime Report Data Stream

Row Number	Name	Year of Birth	Sex	Address	Reported Crime
1	*******	198*	Female	Alma Road, Rosebank	Rape
2	*******	198*	Female	Alma Road, Rosebank	Rape
3	*******	197*	Male	Dikens Road,Salt River	Car Hijacking
4	*******	197*	Male	Dikens Road,Salt River	Burglary

Table 2.2 shows the crime report stream in a sequence that needs to undergo anonymization. In order to achieve k-anonymity, the first step is to remove all the values of the explicit identifiers. Afterwards, the QI, which in this scenario is year of birth, sex and address was identified. On a final note, similar records are grouped into the same cluster such that the minimum number of records in a cluster is k. Table 2.3 is an anonymized version of Table 2.2 using k-anonymity, where k $= 2$ and QI $=$ year of birth, sex, address. Thus, rows 1 and 2 form one cluster while rows 3 and 4 form another cluster in Table 2.3. From Table 2.3, each sequence of values in QI has at least two occurrences. In general, the probability of linking attack occurrence is $\frac{1}{k}$, this means that the probability of a linking attack in Table 2.3 is $\frac{1}{2}$. Thus, the higher the value of k, the lower the probability of linking attacks. In Table 2.3, "$\star$" denotes a suppressed value. Thus, "Year of Birth $=$ 198*" means the year of birth is in the range (1980 - 1989). In this scenario, "1980 - 1989" is the generalized form for the birth-date attribute.

Table 2.2 seems to be a perfect scenario where one could easily cluster records with similar attributes in order to achieve anonymization. However, there could be instances where the range of values of a new record does not fall in any of the clusters; in such a case anonymization will be challenging and difficult to achieve and too much information can consequently be lost.

2.4.1 Classification of k-Anonymity

K-Anonymity can generally be classified into two as depicted in Table 2.4 [101], namely:

- Hierarchy-based generalization
- Hierarchy-free generalization

Table 2.4: Depiction of Classification of k-anonymity

K-Anonymity	Hierarchy-Based Generalization (requires user's input)
	Hierarchy-Free Generalization (does not require user's input)

1. **Hierarchy-Based Generalization**

   In hierarchy-based generalization, the domain of each attribute is usually stated using a hierarchy called domain generalization hierarchy (DGH) [38, 85]. A domain is an acceptable value from which each attribute of a table can be drawn [77]. The acceptable values of each attribute are usually constructed from DGH, which gives rise to the value generalization hierarchy (VGH) [101].

   HBG expects the data analyst or programmer to specify explicitly the DGH and VGH of each attribute before the generalization or anonymization process begins [101]. Algorithms in [38] make use of hierarchy-based generalization.

2. **Hierarchy-Free Generalization**

   This approach uses clustering and partitioning to produce a generalized result. It does not require a user-defined generalization tree or hierarchy like hierarchy-based generalization. Algorithms in [43] uses hierarchy-free generalization that is based on clustering.

While k-anonymity effectively prevents identity disclosure, it is insufficient for the prevention of attribute disclosure. Identity disclosure takes place when an individual is linked to a particular record in the released database, while attribute disclosure occurs when new information about some individuals is revealed [52]. As a result, newer privacy-preserving models have been conceived to address these limitations.

2.4.2 ℓ-Diversity

Machanavajjhala et al [57] determined that k-anonymity has the following weaknesses:

1. Attacks based on background knowledge are not protected by k-anonymity. Such attacks happen as a result of prior knowledge of some additional external information available to the attacker [57].

2. K-anonymity has the tendency to create groups that leak information owing to lack of diversity in the sensitive information. An attribute is sensitive if it contains private information whose value must not be known for any individual in the dataset (e.g. the nature of crime committed by a person, such as rape, murder).

To address these deficiencies of k-anonymity, the concept of ℓ-diversity was introduced into k-anonymity. ℓ-diversity ensures that each equivalence class has at least ℓ "well-represented" values in the sensitive attributes, where $\ell \geq 2$ [52, 57]. Therefore a table is ℓ-diverse if the distribution of a sensitive attribute in each equivalence class has at least ℓ-well represented values. Equivalence class is a set of k-anonymous records that have the same values for the QIs [52]. An equivalence class is considered "well represented" if the following conditions are satisfied [57]:

ℓ-distinct: This is the simplest form of ℓ-diversity. It simply ensures there are at least ℓ distinct values for the sensitive attribute in each equivalence class. Distinct ℓ-diversity does not prevent probabilistic inference attacks. An implication of this attack is that an equivalence class may have one value appearing much more frequently than other values and as a result an adversary could probably come to the conclusion that an entity in the equivalence class is very likely to have that sensitive value that has higher frequency. This motivated the development of the stronger notions of ℓ-diversity described below.

a Probabilistic ℓ-diversity: An anonymized dataset is said to satisfy probabilistic ℓ-diversity if the frequency of a sensitive value in each group is at most $1/\ell$. This implies that an adversary cannot infer the sensitive value of an individual with a probability greater than $1/\ell$. This takes care of the shortcoming of distinct ℓ-diversity .

b Entropy ℓ-diversity: This is the most complex form of ℓ-diversity. The entropy of an equivalence class, E, is defined to be

$Entropy(E) = -\sum s \in Sp(E,s)logp(E,s)$

where S is the domain of the sensitive attribute, and p(E,s) is the fraction of records in E that have sensitive value s. A table is said to have entropy ℓ-diversity if for every equivalence class E,

$Entropy(E) \geq log\ell$. Entropy ℓ-diversity is stronger than distinct ℓ-diversity.

c Recursive (c,ℓ)-diversity: Recursive (c,ℓ)-diversity is a compromise definition of ℓ-diversity that ensures that the most common value does not appear too often, and the less frequent values do not appear too seldomly.

2.4.3 From ℓ-Diversity to t-Closeness

In spite of the attempts made by ℓ-diversity to prevent attribute disclosure, it still has some limitations, which are discussed below:

1. It could be hard to understand and unnecessary to achieve when possible values of a sensitive attribute are widely apart. In order to understand this better, let us assume that the original database has a single sensitive attribute, criminal type, that can take only two possible values: major crime and minor crime. In addition, suppose there are 1000 records, with 99% of them being major offences and 1% being minor ones, these two values have very wide sensitivity degrees. In a situation like this, it is very difficult if not impossible to achieve ℓ-diversity effectively.

2. It is insufficient to prevent attribute disclosure through similarity and skewness attack. A similarity attack occurs if an attacker can learn important information in an equivalence class when sensitive attributes are distinct but similar semantically, while a skewness attack occurs if the distribution of values of the sensitive value within a given equivalence class differs from the distribution of the values for the same attribute over the whole population, i.e. skewness attack occurs when the overall distribution of data in the table is skewed [52].

To solve these weaknesses, the notion of t-closeness was born. t-closeness ensures that the distribution of values of a sensitive attribute(s) in an equivalence class is similar to that of the entire table. Assume the crime type distribution in a table is 35% rape, 40% murder and 25% theft; t-closeness will ensure that a similar ratio holds in each of the equivalence classes. Therefore an equivalence class is said to adhere to t-closeness if the distance between the distribution of a sensitive attribute value in the class and the distribution of the same attribute value in the whole table is no more than a threshold, t, where $0 \leq t \leq 1$. Consequently, a table is said to have t-closeness if all equivalence classes have t-closeness.

To illustrate how t-closeness works, let us assume Table 2.5 is the original table that needs to undergo anonymization and the attribute 'crime type' is sensitive while the attributes age and address are QI. The first step in order to achieve t-closeness is to form an equivalence class(es), which gives us Table 2.6. Then the next step is to find the distribution of sensitive values in each equivalence class and compare it to the overall distribution of the whole table. The probability of rape among the original dataset in Table 2.5 is $1/4 = 0.25$ while the probability of rape among individuals in the first equivalence class is $1/2 = 0.5$. Assuming the value of $t = 0.1$, then Table 2.6 does not satisfy t-closeness because $0.5 - 0.25 > t$ (where $t = 0.1$). For Table 2.6 to satisfy t-closeness in its present form, the t-value must be as high as 0.25. However, if there is a necessity for t-closeness to be satisfied for the t-value of 0.1, then the two equivalence classes will be merged. A disadvantage of this merging process is that higher IL will be incurred.

Table 2.5: Original Crime Report

Row Number	**Name**	**Year of Birth**	**Sex**	**Address**	**Reported Crime**
1	Charity	1990	Female	10 Pillans Road, Rondebosch	Rape
2	Hannah	1996	Female	2 Pillans Road, Rondebosch	Theft
3	Charles	1963	Male	10 Loweryork, Woodstock	Car Hijacking
4	Williams	1965	Male	12 Loweryork, Woodstock	Burglary

Table 2.6: Anonymized Crime Report Data Stream, where $k = 2$ and $t = 0.25$

Row Number	**Name**	**Year of Birth**	**Sex**	**Address**	**Reported Crime**
1	*******	199*	Female	Pillans Road, Rosebank	Rape
2	*******	198*	Female	Pillans Road, Rosebank	Theft
3	*******	197*	Male	Loweryork, Woodstock	Car Hijacking
4	*******	197*	Male	Loweryork, Woodstock	Burglary

Table 2.7: Summary Information on Data Anonymization Techniques (S/N indicates Serial Number).

S/N	Features	Major Approach	Advantage	Limitation
1	Swapping	Interchange of values	Simple to understand	Its output results in an untruthful data-set
2	Additive Noise named noise	Addition of a variable	Fast	Its output results in an untruthful data-set
3	k-anonymity and Suppression	Generalization truthful	Data remains	Prone to background knowledge attack
4	ℓ-diversity	Bayes Optimal	Data remains truthful	Prone to similarity and skewness attack
5	t-closeness	Earth Mover Distance metric	Data remains truthful	Excessive information loss

While the aforementioned techniques have proven useful in diverse ways and domains [52, 57, 83], two important observations are worth noting:

1. Choice of Technique: It is observed that there is no single technique that is entirely perfect, each technique has its strengths and some form of limitation(s). However, the limitations in one technique could be partially or adequately addressed by some other technique. Table 2.7 presents a high-level summary of data anonymization techniques, by presenting the advantages and limitations of each technique. This means that one can leverage the strengths in two or more techniques to achieve desirable and effective anonymization, which is what this research tries to achieve.

2. Data Utility: This simply refers to how useful the data is after it has gone through the anonymization process. It refers to the level of impact the anonymization process has on the quality of the generated pattern, during analysis for knowledge support. While anonymization is potentially useful, it is necessary to pay attention to: (i) what impact a chosen technique could have on the utility of data; and (ii) how to avoid or reduce IL during the anonymization process. Thus, IL is an important consideration in this research.

2.5 New Trends in Privacy Preservation

Differential privacy is a recent privacy-preserving technique that was initially developed to use the interactive setting model as its privacy mechanism [22, 21]. Inherently, there are two natural models for privacy mechanisms, namely interactive and non-interactive [51]. In the non-interactive setting (an example is k-anonymity and variants) the data collector, which is usually a trusted entity, publishes an anonymized version of the collected data, while in the interactive setting an interface is provided through which users may pose queries about the data and are likely to get noisy answers.

Differential privacy is obtained by using a computational mechanism, noise addition [30]. Usually, the real value f(D) of the response to a certain user query f is computed, and then a random noise, say Y (D), is added to mask f (D), that is, a randomized response $\kappa(D) = f(D) + Y(D)$ is returned [79]. Differential privacy attempts to limit the knowledge users can derive from query responses.

Though differential privacy was proposed to use the interactive setting, recent research in [5, 11, 23] has proposed how it can be adapted for non-interactive setting, thus bringing it on par with k-anonymity and its variants with respect to the privacy mechanism model being adopted. Furthermore, recent research has shown that the use of k-anonymity with random sampling [51] or k-anonymity with t-closeness [19] can result into privacy results similar to that of differential privacy. One implication of this is that a dataset that satisfies k-anonymity with t-closeness equally satisfies $\in$-differential privacy [79]. Therefore, in this thesis, k-anonymity and its variants were chosen because of the simplicity [57, 25], effectiveness [84, 81] and high utility [19, 18] it offers.

2.6 Information Loss Metrics

Data anonymization could potentially lead to IL. IL quantifies information that is lost during anonymization. The less the IL incurred, the better the quality and utility of anonymized data. IL metrics measure how much anonymized data differ from the initial or original form [37]. The common metrics are as follows:

1. Precision Metric: This takes the height of generalization hierarchy into consideration during calculation of IL. For a given cell of an anonymized table, the precision metric is calculated by finding the ratio of the cell's generalization level to the total possible generalization levels [85].

2. Discernibility Metric: It penalizes each tuple based on the number of tuples that are indistinguishable from it [4].

3. Classification Metric (CM): It counts those tuples that have class labels different from the majority. CM calculates information loss for data transformation based on the fact that the intended usage of data is for predictive modelling [38].

4. Generalized Loss Metrics: It considers the size of a cluster and the entire data distribution. These metrics calculate IL based on the fact that the intended usage of data is not known at the time of release [38].

While most of the aforementioned privacy-preserving techniques discussed in section 2.4 have been used for anonymization, the focus has mostly been on static data, not data stream. Table 2.8 states the difference between static data and data stream anonymization. In addition to information loss, there is need to consider effective ways of applying anonymization techniques to streaming data.

Table 2.8: Difference between Static Data and Data Stream Anonymization

S/N	Static Data Anonymization	Data Stream Anonymization
1	It does not require real-time processing	It requires a real-time processing
2	The fastest approach for obtaining an approximate solution is in polynomial time	Processing time should not be more than O($\|S\|$), which is linear to data stream size, S
3	It requires multiple scans in order to achieve generalization with the least IL	Multiple scans of data are not possible because data flows at high speed and only one scan is possible

2.7 Data Stream Anonymization Concept

In the last decade, many privacy-preserving techniques such as k-anonymity and other complementary privacy-preserving techniques have emerged to encourage users and data holders to share and release information without fear of data disclosure. However, all of these techniques were conceptualized for

static data and cannot be directly applied to continuous or flowing data (data streams) because of the following reasons [29, 10, 52, 101, 102]:

1. **Temporal Dimension**: Data streams have a temporal dimension, i.e. there is a maximum acceptable delay between inflowing data and its corresponding anonymized output. More often than not, the anonymized output triggers other actions, such as knowledge generation. Hence, the receiving application should have strong guarantees on the maximum delay of its input data.
2. **Transient**: The technique assumes that data are static, but data streams are continuous and transient in nature [94].

In light of the aforementioned factors that could hinder the successful application of anonymization techniques on streaming data, it becomes necessary to come up with useful principles that can guide effective anonymization of streaming data.

2.8 Data Stream Anonymization

Guo and Zhang [102] came up with the following fundamental principles in designing an anonymization scheme for data streams:

1. Scanning of data should occur only once and the time complexity for data stream anonymization should not be more than (O|S|), which is linear to the data stream size, S.
2. Just like clustering of a data stream, anonymization of a data stream can be divided into online and offline. The online division scans the records in the stream once and stores them in a buffer. The offline division carries out the anonymization using the records in the buffer.
3. A threshold should be set for the following parameters: size of buffer, waiting time of a tuple (delay constraint), reusable k-anonymized cluster set and k-anonymized cluster. The size of a buffer depicts the total number of tuples under consideration for anonymization at a particular time. The delay constraint specifies how long a tuple can stay in the buffer. The delay constraint could be count-based and/or time-based. A re-usable k-anonymized cluster set is a collection of clusters that has successfully output some anonymized tuples and kept for re-use. Any subsequent record that fits into any of the clusters in the reusable k-anonymized cluster set can be output

immediately without going through k-anonymization because the cluster has already satisfied k-anonymity previously. A k-anonymized cluster is a cluster that will satisfy or has satisfied k-anonymization requirements.

4. The space occupied by the data streams anonymization scheme should be constrained. During anonymization of data streams, most of the space is used for storing tuples that have arrived and k-anonymized clusters. There should consequently not be infinite growth of space occupation.

5. Published k-anonymized clusters should be reusable and whatever reusable strategy is employed should be as simple as possible. According to [29], a single k-anonymized cluster set is adequate to achieve effective cluster reuse.

Figure 2.6 shows how data stream anonymization takes place.

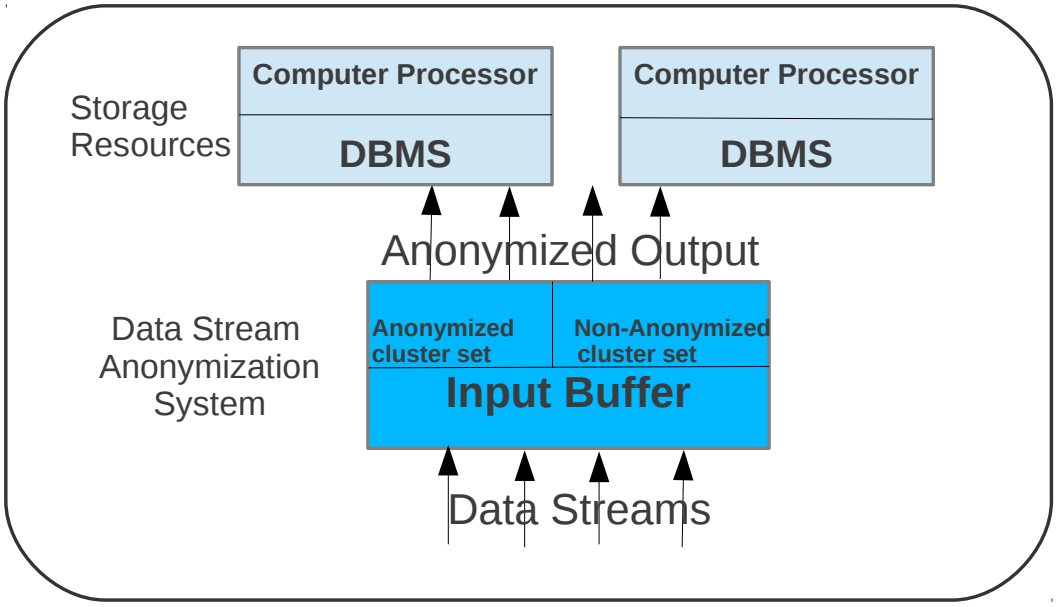


Figure 2.6: Diagrammatic Sketch of how Anonymization Technique can be Applied to Data Stream

From the literature [49, 102], anonymization of data streams occurs using either the perturbative or the non-perturbative methods.

2.8.1 Perturbative Method

To the best of the researcher's knowledge, only one documented work by Li et al. [49] achieves data streams anonymization using the perturbative method. Their approach achieves privacy of streaming

data by modifying the values of incoming data through the addition of noise. That is, additive noise, $E \in R_N^T$ is added to the record, where E denotes random noise, N is the number of streams and T is the current timestamp. Each E_{ti} is the noise added to the i^{th} stream at time t. Therefore, the anonymized (perturbed) stream, A^*, can be expressed as: $A^* = A + E$.

A major limitation of this work [49] is that the anonymized data becomes too difficult to analyse for knowledge support, as a result of too much artificial noise. Furthermore, the resulting anonymized data, A^*, can only effectively handle numeric data.

2.8.2 Non-Perturbative Method

Unlike the perturbative anonymization method, the non-perturbative method takes the semantics of the data to be anonymized into consideration by using category generalization [32, 85, 4, 54]. Specifically, the non-perturbative method works by replacing a specific value with a more general one based on a hierarchical or generalization tree. As a result, the number of distinct tuples in the data set is reduced and, therefore, the level of anonymity increases [18].

Existing data stream anonymization schemes using k-anonymity and/or ℓ-diversity of the non-perturbative method can be categorized as belonging to either hierarchy-based generalization or hierarchy-free generalization.

2.8.3 Hierarchy-Based Generalization of Data Stream Anonymization

Data stream anonymization in this category uses hierarchical trees (tree structures) to achieve anonymization. Examples of data stream anonymization schemes that adopts hierarchy-based generalization are Stream K-anonYmity (SKY) [50], Sliding Window Anonymization Framework (SWAF) [94] and K-AnonymIzation Data Stream Base on a Sliding Window (KIDS) [102].

1. **Stream K-Anonymity**

   SKY represent some of the pioneer work in literature that considers the use of non-perturbative method (k-anonymity) for data stream anonymization [50]. SKY requires the following input parameters: Data stream, specialization tree, *k* (k-anonymity value) and delay constraint. A data stream is simply the set of inflowing data that needs to undergo anonymization. Each QI attribute,

has a predefined specialization tree. The specialization tree is a directed tree, where each node is a vector from which an acceptable value is drawn.

When SKY reads a record from the stream, it searches the specialization tree to find the most specific node that generalizes the new record. SKY's specialization tree nodes can either be "candidate" or "work". Candidate nodes are nodes that are yet to satisfy k-anonymity, while work nodes are nodes that have satisfied k-anonymity and have been kept for future re-use. Consequent to this, when a new tuple arrives in the stream, SKY determines the best node to place the new tuple. If SKY places it in a "work" node, it will be output immediately. If otherwise (i.e. in a candidate node), it may not be output until the node has satisfied either k-anonymity and/or delay constraint. The following are the observed limitations in the SKY technique:

(a) It has no criteria for the choice of re-usable (working) nodes. As a result, a cluster that resulted in high IL may be kept for re-use.

(b) SKY's time and space complexity, which are O($|S|\delta log\delta|S|$) and O($|S|$) respectively, are too high and unacceptable for data stream anonymization [102].

(c) Its use of a generalization tree for the anonymization of numerical values makes the process of finding a suitable hierarchy difficult [101].

2. **Sliding Window Anonymization Framework:**

The SWAF technique incorporates a sliding window on data streams [94]. The sliding window contains the most recent part of the stream. As new tuples arrive in the stream, SWAF updates the stream by replacing the oldest ones in the sliding window. The technique also makes use of a specialization tree. In the initial stage, the sliding window behaves as a static data set, and SWAF runs an heuristic algorithm on the window in order to generate a specialization tree. As the sliding window is being updated, SWAF uses another heuristic algorithm to continuously adjust the specialization tree. The technique ultimately obtains a k-anonymization for the sliding window from the specialization tree. However, the following are the limitations of the technique:

(a) The time and space complexity is too high for data streams, as revealed in [102].

(b) Its use of a specialization tree for anonymizing numerical values makes the process of finding a suitable hierarchy difficult [101].

(c) It has no restriction on the maximum number of records that can form a k-anonymized node. For instance, if *k* is set to 50, there is a possibility that a node can overshoot that and have about 70 records in the node.

3. **K-Anonymization Data Stream Based on a Sliding Window:**

The KIDS technique is similar to SWAF; it also makes use of a sliding window. One of the major contributions of KIDS is the incorporation of distribution density [102]. With the use of the distribution density parameter, successive data streams can be predicted so that a leaf node in the current stream with a high distribution density prediction is set aside for future re-use. Ultimately, this helps to reduce IL. KIDS consist of two parts, namely: (i) tree construction and; (ii) tree update (adjustment).

- Tree Construction: This is the initial stage of the anonymization process, at this point, data in the sliding window are static. The root of the tree is the most general value of all QI. For example, all tuples of a sliding window may be generalized to root node (e.g. Student: for student level; South Africa for residential location, where student level and residential location are QI). The construction of other nodes emerges from the root.
- Adjustment of the Tree: When a new tuple arrives in the sliding window, it is generalized into the most specific node, n_i, of the specialization tree that the new tuple can fit into. Then the tree updates in one of the following two situations:
  a. If the node, n_i, can be further specialized as a result of the new tuple and the child node is not violated. If this is possible, then the node splits.
  b. If the node, n_i, is frozen and contains k-1 tuples, then the tuples will be released as a result of the new tuple. A frozen node contains tuples less than k or has IL higher than a certain threshold.

The following are the shortcomings of the KIDS method:

(a) The time and space complexity is too high for data streams [102].

(b) Its use of a specialization tree for anonymizing numerical values makes the process of finding a suitable hierarchy difficult [101].

2.8.4 Hierarchy-Free Generalization of Data Stream Anonymization

Data stream anonymization algorithms in this category uses a clustering method to achieve anonymization. Clustering operates by grouping similar objects together so that those in each cluster are more similar to each other than to objects in other clusters. Examples are continuously anonymizing data via adaptive clustering (CASTLE) [10], B-continuously anonymizing data via adaptive clustering (B-CASTLE) [93], fast anonymizing algorithm for numerical streaming data (FAANST) [100], delay-sensitive FAANST [101], fast clustering-based anonymization of data streams (FADS) [102].

1. **Continously Anonymizing Streaming Data via Adaptive Clustering (CASTLE)**

   CASTLE is one of the pioneer data stream anonymization algorithms that integrates the concept of k-anonymity and ℓ-diversity [10] into data stream anonymization. It usually takes in three input parameters, which are k, δ and β. K is the value for k-anonymity, δ is a threshold for the maximum publishing delay deadline and β is the maximum number of permitted clusters [94]. CASTLE maintains two sets of clusters, namely: k-anonymized clusters and non-ks-anonymized clusters. The k-anonymized clusters are clusters kept for future re-use because their tuples satisfy k-anonymity and have been output. The non-k-anonymized clusters are those whose tuples are yet to expire and have not been output.

   At the initial stage of the anonymization process, there are no clusters in memory [10].Therefore, the first record CASTLE receives in the stream forms a cluster. For subsequent arriving records, CASTLE determines the best cluster in which to place them, among the existing ones. On the other hand, it could happen that no existing cluster can accommodate the new record. In such a case, one of the existing clusters, say T, undergoes enlargement based on the range of interval of the QI attributes of a tuple (t). For example, a cluster T[60-65, Port Elizabeth - a city in the Eastern Cape)] can be expanded to T[60-70, Eastern Cape - a province in South Africa] to accommodate t(68, East London - another city in the Eastern Cape).

   Enlargement of a cluster indicates a greater loss of information. To minimize loss of information, CASTLE chooses the cluster that calls for the smallest enlargement. Furthermore, CASTLE ensures that the IL that may result from a cluster is less than a predefined threshold. If the enlargement of a cluster will warrant IL to exceed the threshold, the new record forms a new cluster.

   CASTLE outputs/publishes its anonymized data in two ways. Firstly, a cluster of a size equal to

or greater than k, is output if it contains any expiring record. Secondly, if a cluster less than k contains an expiring record, CASTLE checks for a cluster that requires the least enlargement for merging so that all records in the resultant cluster can be k or greater than k. The records in the resultant cluster can then be output. If the IL of a cluster whose tuples have already been output fits within an acceptable threshold, this cluster will be stored for subsequent re-use. Such clusters are k-anonymized clusters.

CASTLE enforces ℓ-diversity by ensuring that all records belonging to the same QI group have at least ℓ distinct values for the sensitive attribute. CASTLE considers only a single sensitive attribute for ℓ-diversity. The shortcomings of CASTLE are as follows:

(a) Tassa and Gudes [87] noted that local recoding of generalization leads to less IL. CASTLE uses global recoding. A better approach will be to consider local recoding. Local recoding is a cell-level generalization of each tuple [37]. To illustrate local recoding, assume a specific year of birth, "1984", appears in several records, it may be left unchanged in some, or generalized to 198* in some others, or totally suppressed in other remaining records.

(b) Whenever a record in a cluster of a size lower than k has expired, CASTLE attempts to look for neighboring clusters to merge. Instead of merging the clusters, a better approach might be to remove only the expiring record and retain the other records.

(c) CASTLE does not restrict the size of a cluster. It also does not have a specification for the possible number of initial and final clusters.

(d) CASTLE verifies whether every tuple fits into all available clusters in order to select the one with the least IL. This verification time increases with $|S|$ reaching a time complexity of O ($|S^2|$), which is too high [29].

(e) CASTLE checks for an expiring tuple on the arrival of a new one. A better approach could be the activation of an automatic alert whenever a tuple is due or about to expire.

(f) CASTLE does not place a limit on the maximum number of tuples that a cluster can have. As a result, some clusters have more tuples than others and this may lead to high IL.

(g) The merge operation of CASTLE re-clusters all tuples without considering the distribution of data in the stream. This merging operation further splits any cluster with more than $2k$ tuples. This results in higher IL and an increase in the complexity as a result of the time spent in the splitting process.

2. **B-Continously Anonymizing Streaming Data via Adaptive Clustering**

   In a bid to address some of the limitations of CASTLE, the B-CASTLE technique [93] was developed. It attempts to improve CASTLE as follows:

   (a) To solve the limitation of not placing a limit on the maximum number of tuples that a cluster can have, B-CASTLE introduced a threshold, α, to the cluster size. This ensures that no cluster has more than α tuples.

   (b) In solving the problem associated with the merging operation of CASTLE, which re-clusters all tuples without considering the distribution of data in the stream, B-CASTLE merges a cluster of a size less than k (if it contains an expiring or expired tuple) with its nearest cluster. This process is performed recursively until the resultant cluster has more than k tuples.

   While B-CASTLE is an advancement on the CASTLE technique, it does not place any restriction on the growth size of its reusable cluster set. As a result, its time and space complexity is the same as that of CASTLE [102].

3. **Fast Anonymizing Algorithm for Numerical STreaming data (FAANST)**

   FAANST works in a way different from CASTLE and B-CASTLE by delaying the start of the anonymization process until the buffer is full [70]. Therefore, the main way by which FAANST differs from CASTLE is that it releases the anonymized data stream at intervals rather than continuously [100]. FAANST operates in two phases [100], which are explained as follows:

   (a) Phase 1: This is the first time the algorithm runs. As soon as the number of tuples in the buffer has reached its maximum limit, FAANST partitions tuples in the buffer into different cluster(s) using a k-means clustering algorithm. Any cluster whose size is at least k will be output irrespective of its IL. Only clusters whose IL is not more than a certain threshold are set aside for future re-use. The algorithm waits again to have its memory filled up before processing tuples.

   (b) Phase 2: The second phase occurs when at least one round has executed (that is, from the second round to the last round). A round simply connotes what happens when the processing window (buffer) has reached its maximum limit. When the window gets to its maximum limit, tuples falling into one of the accepted clusters are output while other tuples are partitioned into k' using k-means.

The weaknesses of FAANST are as follows:

(a) Some tuples may stay in the system for too long and expire before output.

(b) The duration of a tuple in the approach was not considered in the IL metric.

4. **Delay-Sensitive Approaches for Anonymizing Numerical Streaming Data (Delay-Sensitive FAANST)**

Zakerzadeh and Osborn [101] came up with a delay-sensitive FAANST as a result of the deficiency of their previous algorithm, FAANST [101].

To solve the limitations of FAANST, a delay sensitive approach [101] introduces some mechanisms namely, (i) a proactive solution, (ii) a passive solution and (iii) a delay parameter. The purpose of these mechanisms is to ensure tuples do not expire.

- Passive solution: This saves the arrival time of each tuples. At the end of each round, it checks for expired tuples. Those that have exceeded the deadline are suppressed and output.
- Proactive solution: This approach has an extra attribute for storing the time of the last visit for each tuple. The initial time of every tuple is its arrival time. A simple heuristic calculated by (current time - TimeOfLastVisit) determines if a tuple can still remain in the system and not expire before the next round. If the output of the heuristic calculation shows that a tuple will exceed its delay threshold, then it will be output with the present round. Otherwise, it retains such a tuple for the next round.
- Incorporation of delay parameter into IL.

The following are the limitations associated with delay-sensitive FAANST:

(a) Tuples in a cluster that satisfies k-anonymity are output irrespective of their IL. There might be a need to study how those clusters that have very high IL can undergo improvement such that the privacy and delay constraint is satisfied.

(b) It does not incorporate new privacy techniques such as ℓ-diversity and t-closeness.

(c) A passive solution leads to more execution time as a result of checking to see if a tuple has exceeded its deadline.

5. **Fast Clustering-Based Anonymization for Data Streams (FADS)**

FADS [102] takes four parameters as input in order to achieve data stream anonymization. The parameters are data stream (S), the k-anonymity requirement (k), the delay constraint (which should be greater than k and the reuse constraint (Tkc). Tkc imposes a constraint on the re-usable clusters. $setkc$ is a set that contains all the reusable clusters. The FADS process is as follows:

(a) It reads a tuple from S and stores it into buffer, Settp, during each round.

(b) The range of numeric attributes is updated as a new tuple arrives

(c) If some tuple, t, is ripe for release, FADS refreshes the re-usable cluster set by removing the k-anonymized cluster that has existed for more than Tkc.

(d) t is then released.

(e) When no more tuples arrive, the remaining tuples will be released. If the number of tuples in a buffer is less than k, it is impossible for anonymization to take place. In such a case, the expiring tuple(s) is/are suppressed or output with a re-usable cluster.

As an advancement, the FADS technique incorporates ℓ-diversity in its approach. To conform to ℓ-diversity, the sensitive attributes of tuples in a cluster should have at least ℓ well-represented values. In other words, the probability of associating a tuple to a sensitive value is at most $\frac{1}{\ell}$. ℓ is the total number of values for a sensitive attribute. In a cluster, the number of tuples with the same sensitive value should not be greater than $\lfloor |C| \div \ell \rfloor$, C is the cluster containing the tuples. Since the size of a cluster should not be less than k, the initial threshold for the number of tuples with a corresponding sensitive value can be set to $\lfloor k \div \ell \rfloor$. If t is chosen to be published with a reusable k-anonymised cluster in $setkc$, it should be checked if the insertion of t will make sensitive values correspond to more than $\lfloor |C| \div \ell \rfloor$.

The limitations of FADS are as follows:

(a) There is a possibility of record expiration.

(b) The scheme does not incorporate t-closeness.

(c) The clustering approach adopted in FADS may release a newly arrived tuple early before its time limit just because some related tuples are ready for publication. This may lead to an

additional IL particularly if it is possible that the tuple could have been released in future with a cluster that has lower IL.

(d) The idea of using suppression when tuples ready for publication are less than k can cause a greater loss of information.

6. **Fast Anonymization of Big Data Streams (FAST)**

Fast anonymization of big data streams (FAST) was born out of the weaknesses of the FADS algorithm. One of the major weaknesses of FADS is that there is a tendency for some tuples to remain in the system for a long time and therefore, are released after a specified threshold comes to an end. It is worth noting that this weakness violates the real-time condition of a data stream application and ultimately increases the cost metric.

FAST uses a proactive time-expiration heuristic to handle the aforementioned challenge. In order for the proactive heuristic to work effectively, a new parameter, expiration time, which represents the maximum delay that is tolerable is introduced. The heuristic works by using a simple formula $(CurrentTime - ArrivalTime) + EstimatedRoundTime \leq ExpirationTime$ to check if a tuple will expire if considered for the next round. This is a major improvement over FADS because FADS does not verify if a tuple can still remain in the system or not, this explains why some tuples are published after expiration.

Table 2.9 presents a high-level summary of information on the data stream anonymization techniques. The approach presented in this thesis attempts to leverage the strength of existing techniques to achieve anonymization, while augmenting with appropriate heuristic and technique for improved performance. In particular, the ℓ-diversity and t-closeness approach were employed to augment the adaptive K-anonymization technique, which is supported using Poisson distribution in order to predict the rate of flow of data in the stream and ultimately reduce IL and enhance the speed of anonymization. Moreover, the use of a three-tier user-defined privacy mechanism for anonymization was considered. The three-tier privacy level preference approach includes low, neutral (medium) and high levels. The choice of three tiers is based on the outcome of previous research that users have varying levels of privacy and can realistically choose between three levels of privacy [13, 72, 97]. The integrated functionality of the approach considered in this research is implemented in a testbed called CryHelp. More details on the approach adopted and CryHelp follow in succeeding chapters.

Table 2.9: Summary of Information on Data Stream Anonymization Techniques

S/N	Features	Major Approach	Advantage	Limitations
1	Li et al [49]	Additive Noise	-	Too difficult to analyse
2	SKY [50]	k-anonymity	Preserves data truthfulness	Time and space complexity is too high
3	SWAF [94]	K-Anonymity	Small memory requirement	Susceptible to homogeneity attack
4	KIDS [102]	K-Anonymity	Incorporates distribution density	Difficult to find a suitable hierarchy
5	CASTLE [10]	K-Anonymity and l-diversity	Ensures freshness of anonymized data	High Information loss
6	B-CASTLE [93]	K-Anonymity	Faster merge operation	High information loss
7	FAANST [100]	K-Anonymity	Batches Anonymization Processing	Can only handle numeric values
8	Delay-Sensitive FAANST [101]	K-Anonymity	Delay handling	Can only handle numeric values
9	FADS [29]	K-Anonymity and l-diversity	Low time complexity	Record expiration and does not incorporate t-closeness
10	FAST [59]	K-Anonymity	Delay handling	Does not incorporate dynamism into its delay handling

2.9 Contributions and Chapter Summary

This chapter began with a motivation for data privacy and presented data anonymization as a viable solution to achieve data privacy, among other existing or conventional solutions. In particular, the limitations of access control and cryptography in achieving data privacy are discussed, followed by the rationale for data anonymization. Furthermore, a critical review of conventional and existing state of the art approaches and techniques in data anonymization was presented. These were achieved

through various expository illustrations and tabular representations. The potential usefulness as well as shortcomings of these approaches were discussed, while emphasizing the lack of attention to the approaches in data stream anonymization. The contribution of this research in addressing some of the limitations in existing techniques is discussed. This research attempts to address the limitations related to record expiration caused by not considering the dynamism of the data stream; high IL caused by not considering the distribution of the present and future data stream and in-balance privacy caused by not taking users' privacy preferences into consideration. This has led to the proposition of an adaptive anonymization technique augmented with Poisson probability distribution and a three-tier level framework for achieving user-preferred personalizing preference.

Poisson probability distribution was incorporated into the stream in order to improve on the reduced record expiration and IL by studying the rate at which data flow in the stream and the rate at which data will flow in the future stream.The limitation of user-preferred personalized privacy was addressed by first conducting a survey in the form of a questionnaire in order to ascertain users' preference. Responses from the questionnaire were evaluated and then modeled into the proposed anonymization framework in the form of a three-tiered privacy level. A three-tier privacy level preference approach consists of low, neutral (medium) and high levels. Detailed documentation on the contribution of this study is discussed in subsequent chapters. Also,further information about the framework follows in subsequent chapters.

Chapter 3

Data Anonymization Framework

This chapter presents a general framework for the research work carried out in this study. In order to have a realistic and detailed framework, a detailed system requirements survey was conducted using standard means of data collection and analysis. The discussion in this chapter first reports on the overall framework of the research in section 3.1. Afterwards, in section 3.2, the detailed approach used for the systems requirement of the user layer, its design and implementation was explained. Section 3.3 discusses how the tool for data collection (i.e. CryHelp App) evolved through different prototypes. Section 3.3.3 discusses the design of the emergency report. Finally section 3.4 discusses the CryHelp Application in detail.

3.1 General Framework

Generally, research aims to contribute to the body of knowledge through various means of experimentation, investigation and observation, among others [47]. A research problem in this study is that when records are time-sensitive or need to be processed in real time, the undue delay of the records result in high levels of IL from dropped records, which is undesirable. In addition, there was a need to identify the sliding windows or data streams that are most appropriate to anonymize data such that privacy is preserved with low IL. Lastly it was observed that existing anonymization schemes are structured to accept a static or constant anonymization value for an entire dataset, thus enhancing data privacy, but have the drawback of not being practical for use in real life situations. As such, the objective of this study is to develop a test-bed framework to preserve privacy during real-time information sharing of

crime reports such that the solution is usable in real life and minimizes IL. For this reason, the study uses experimental methods and techniques of empirical enquiry, as well as a quantitative research method. Quantitative research focuses on numeric values and collects data using methods such as questionnaires and experiments [47].

Furthermore, in order to understand the crime-reporting process and the users of the system at the user layer, a survey and interview were carried out among respondents who participated in the process of reporting a mock crime in order to collect data. The respondents included users who had been victims of crime and did not report the crime, users who had been victims of crime and had reported the crimes, and users who had never been victims of crime. The choice of respondents who participated in the study was based on Nielsen's suggestion to "identify a design's most important usability problems, rather than run a big and expensive study" [64, 66, 65].

The conceptual framework, which is divided into four layers, namely a: user layer, network layer, algorithmic layer and application layer, is depicted in Figure 3.1. The user layer captures users' input which could generally be achieved through the use of mobile devices such as mobile phones, Personal Digital Assistants (PDA) and laptops. In short, the user layer is responsible for data collection from the user. In this research, the focus was on how to capture input using a mobile phone. The basis for focusing on mobile phone is because of the success of previous studies [17, 7, 41] which affirm that the use of mobile phones aid in efficiently or securely reporting a crime. These inputs are transmitted through a network. The network layer is responsible for the transmission of users' input into the algorithmic layer. At the algorithmic layer, these input data are buffered for the purpose of effective real-time anonymization. In crime-reporting applications, where real-time decision making is crucial; the buffering and anonymization scheme must be delay-sensitive to avoid any negative effects on real-time publication of anonymized data. At the application layer, the anonymized data are analysed by third-party service providers (such as data miners, crime analysts) for crime pattern identification. Any knowledge of crime trends or patterns that has been derived can then be reviewed by security agencies (policy makers) and used for different safety and situation management decisions.

This research only focuses on the user and algorithmic layer of the overall framework. The details about the design of the user layer is explained in the remaining part of this section, while those of the algorithmic layer are explained in subsequent chapters. The algorithmic layer uses two techniques: Poisson probability distribution and a three-tier personalized privacy scheme to support its anonymization process as shown in Figure 3.2. More detail about how this is achieved is documented in Chapters 4

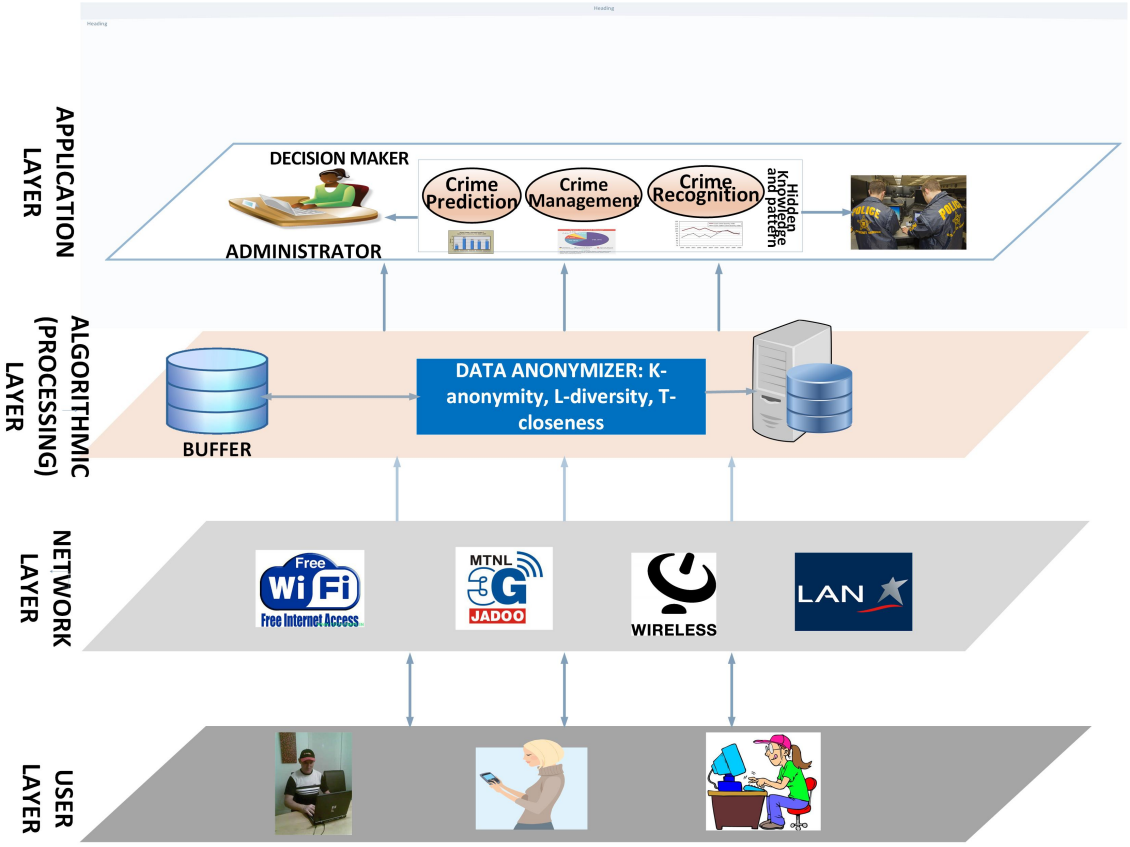


Figure 3.1: Depiction of the Conceptual Framework of the System

and 5.

3.2 User Layer: CryHelp

As indicated in previous chapters, this research considers the crime domain as an application scenario for achieving data stream anonymization. However, the ideas in this research extend to any other domain that requires anonymization of sensitive data. Thus, as a practical means to achieve anonymization in the crime domain, a crime-reporting application named CryHelp, was developed. Following is a discussion of further details on the CryHelp App as applied in this research.

As reflected in the user layer in Figure 3.1 and in the specific case of the CryHelp system, the user makes use of a mobile device that runs an android operating system (OS) to report crime incidents. The android OS is considered because it is proven to be a successful platform for mobile apps and it has an 85% market share among smartphone users [9, 88]. CryHelp enables users to create and effectively

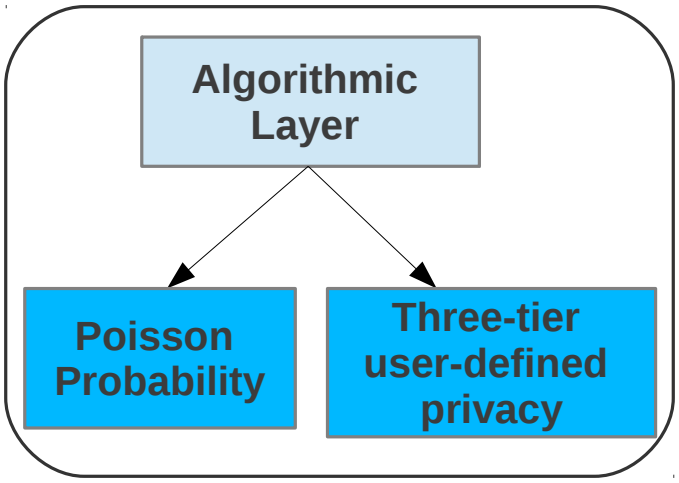


Figure 3.2: Techniques Used for Supporting Anonymization

fill in crime reports that are a replica of the existing paper-based crime report of the University of Cape Town (UCT). The UCT paper-based means of reporting was used as a model for CryHelp for two key reasons:

- CryHelp could serve as an electronic means of crime reporting at the university.
- The students at the university served as the primary source of data collection for demonstrating the applicability and effectiveness of the approach in achieving anonymization.

3.2.1 Requirement Analysis

Requirement analysis was primarily carried out using interviews, questionnaires and group discussions with students and stakeholders. Key stakeholders interviewed to identify these requirements were police officers, crime investigation officer(s), victims of crime, researchers and interface designer. The requirements were integrated into the process of CryHelp App development.

The requirement analysis of the user layer is classified into two, namely functional requirements and non-functional requirements. Functional requirements define the function of the system while non-functional

requirements specify the scope of the system, that is how the functions of the system should be carried out. The functional requirements of the CryHelp App are classified into two namely: (i) **standard crime reports**; and (ii) **emergency**.

The standard crime reports enable end-users to use a mobile device to carry out the following functions:

1. Save the user's personal information.
2. Create a substantial crime report.
3. Send crime reports.

The emergency crime report function enables users to use a mobile device for the following:

1. A single input that enables prior collection of data.
2. A response (confirmation) message when a report is successfully sent.

The non-functional requirements are classified into two: (i) **the platform on which the system is required to run**; and (ii) **the format storage for the input**. The CryHelp system is required to run on an android device platform. Hence, the system stores user data in an XML file. XML was chosen as the target output because of the uniformity of the format. In the next subsection, user requirements were explained.

3.2.2 User Requirements

Based on the recommendation of [16], 20 users were consulted to determine user and design requirements. The sample contained users who have had experience in user design interfaces, had been affected by crime previously and other users who had never been personally affected by crime. The requirements were gathered through interviewing and brain-storming. Table 3.1 presents a summary of the user requirements. The requirements were also augmented by the interview sessions carried out before and after prototype interaction.

A key observation emanating from lines 2 and 6 of Table 3.1 is the fact that some users or crime victims had concerns about their personal identity and adopted a privacy policy. It is also noted that

Table 3.1: User Requirements

Serial No	**Requirements**	**Description**
1	Ease of usage and short input time	Users wanted an application that does not require a lengthy process or time during crime reporting. This implies that the interface must make simple tasks quick to perform. For instance, a user suggested the use of check boxes for yes/no responses.
2	Anonymity: Privacy for sensitive crimes	Users wanted to be able to report some crimes without having these traced back to them.
3	Continuity and re-usability	The application must be able to save an incomplete report so that users can complete it later. Users did not like the idea of having to restart an earlier incomplete reporting process all over again.
4	Replicability and reliability	Users want to be sure they will have the same experience as when reporting in-person. They also want a guarantee that their report is being considered.
5	Simplicity	Users want an application that is simple to use and requires little or no assistance for effective usage. In other words, users do not want the effective use of the application to depend solely on user documentation.
6	Security and data privacy	Some users wanted the privacy of the application to be user-centric while others wanted an assurance that their personal information would be secure.

many factors influence how users want their privacy protected. For instance, some users wanted the protection of their identity to be based on the sensitivity of reported crime. Furthermore, they seemed to want to be aware of the privacy scheme implemented by the law enforcement authorities to guide their decisions on privacy, in terms of releasing their personal identity (sensitive) information. Hence, CryHelp incorporates functionalities that allow some level of control and meeting of user requirements in this regard. Having determined users' requirements, the next milestone was to develop a prototype.

3.3 Prototypes

Prototyping can be simply defined as building a scaled-down version of the desired system [90, 92]. A prototype provides a platform for users to test a tangible component of the system and also a means by which users' input can be gathered for prototype refinement. Through the use of prototypes in this research, the user interface design of the CryHelp App was able to undergo criticism and evaluation. The criticism and evaluation conform to the standard classification of prototyping as either high or low [27, 39], based on the fidelity with which prototypes resemble the original system in terms of design, timing, and interaction. In addition, criticism and evaluation are meant to improve the usability of the overall CryHelp App prototype.

Figure 3.3 shows the iterative cycle involved in the prototype design. The design phase involves interacting with the end-users to have a blueprint of the CryHelp application, while the implementation cycle involves the way in which the output from the design stage was executed and finally the evaluation phase involves interacting with the end-users again to see if their input has been well represented.

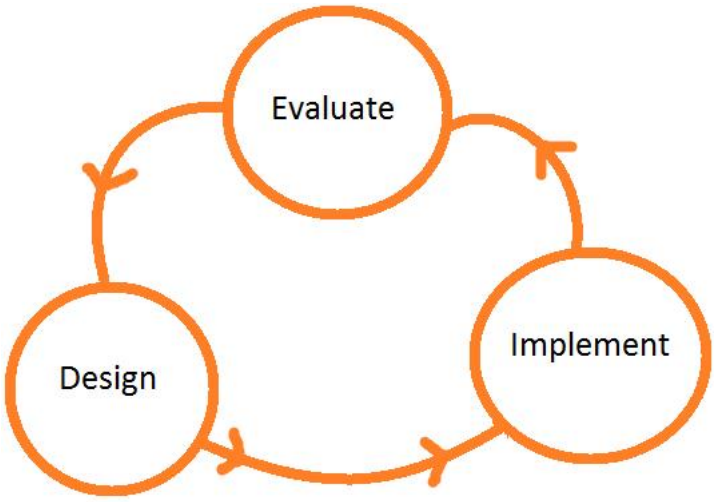


Figure 3.3: Depiction of the Iterative Design Cycle Used during Prototyping

3.3.1 Low Fidelity Prototype (Paper Prototype)

Low fidelity prototypes are easy and quick to implement, allowing immediate testing of a concept [76]. They are also low-cost and allow design ideas to be conceptualized at the early stage of application development. In this research, paper-based prototypes were used for the low fidelity prototyping and it

has been identified as a valuable approach [27]. These prototypes were not only cheap to produce, but also allowed users to manipulate the (potential) CryHelp features more freely with better and honest feedback. Furthermore, users could participate without having to use any computer skills or a computer at all. This allowed user to try out the interaction design rather than the visual design.

For effective and fruitful design outcomes, two iterations for the paper prototype were used. In the first iteration, there were two major goals:

- Get users to recognize what crime reporting is all about.
- Involve users in helping to design an interface that is a logical extension of what they understand a crime report to be.

Having explained these goals to the participants, the prospective users were allowed to give design input on the full crime-reporting application. Some of their input is illustrated in Figure 3.4. During the compilation of their inputs, similar design ideas were grouped together and given preference over less common ideas. The first iteration resulted in a prototype with three major components, namely the main screen, user details form and full report. The main screen, which is the home page, contains two buttons that allow the two other tasks (user details and full report) to be launched and completed.

The second iteration involved users interacting with the final outcome of the first iteration. Comments were again received from end-users and these brought about the following significant changes:

1. The user details button was completely removed. The essence of this was to ensure that details of users are not prompted for each time the application is launched or used. As a result, users are only prompted for their details if they are using the application for the first time. This means that repeat users have their details saved and therefore do not need to enter their details.
2. A new field was also added to handle privacy. The essence of this new privacy field was to allow users to indicate how their data would be anonymized before sharing it with a third party.
3. The full report remained unchanged.

Paper prototypes have the disadvantage of not allowing the implementation of numerous features such as animation and gesture input, which is why high fidelity is adopted.

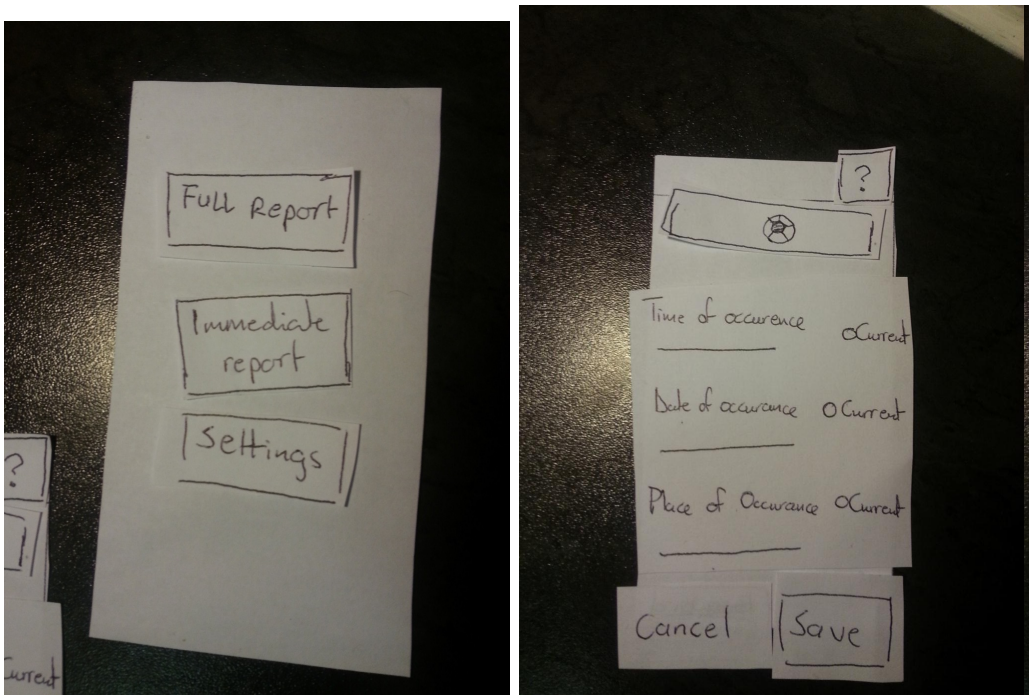


Figure 3.4: Sample of Paper (Low-Fidelity) Prototype Images

3.3.2 High Fidelity Prototype

High fidelity prototyping was used when the basic interface model had been fleshed out. This entailed using a GUI builder to create a click dummy that was a true replica of the final system, but did not necessarily provide any functionality. At this level, the CryHelp prototype demanded more involvement and was designed to work on the target device. Figure 3.5 is an illustration of the high fidelity prototype. Moreover, end-users could gain a sense of what the CryHelp system was realistically about and comment on what they considered good, what needed modification or additional fields considered necessary. Furthermore, end-users could discover how to use the system at this level.

Moreover, at this level of prototyping, it became clear that end-users wanted to see or have some features on CryHelp that allow them have control over their personal information. A recommendation on emergency reporting design was made as well.

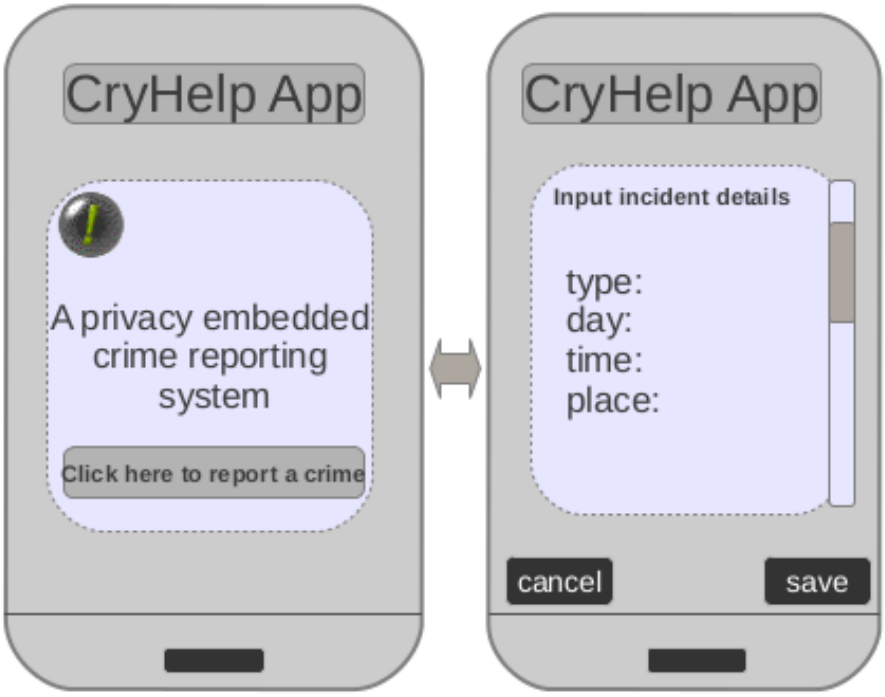


Figure 3.5: Depiction of High-Fidelity Prototype with Dummy Click

3.3.3 Emergency Reporting Design

As a result of constraints in time, the full crime-reporting interface was designed with the emergency reporting interface. In order to source suggestions, the intended users of the system were involved. In the course of the design, users were asked to brainstorm on how they would like to interact with their mobile device and the CryHelp App during emergency crime reporting. Some of the suggestions given included being able to report any given crime using the device in a terrifying situation, being able to scribble letters on the touch surface of the device while simultaneously reporting the crime or simply pressing a button on the device in a manner similar to E9 [69]. E9 is a service offering a means of immediately reporting crime via the use of speed dial. The service is targeted at institutions of tertiary education such as UCT, which have an interest in protecting their affiliates and campus premises.

Having gained a good grasp of how the CryHelp App should look like in real life, the application was then implemented.

3.4 Implementation Environment

The target device for the final solution is an android device. The particular device used is the Samsung Galaxy S3 running android 4.2. In order to meet the specifications of the android device the following environment was used:

Software Development Kit

In order to allow effective implementation of both the interface and communication components, the JAVA programming language was used. For the development of the interface the eclipse Software Development Kit for android development, a tool made available by Google, was used. The target application program interface for the application was the latest android available at the time of development, Android 4.3 Jelly bean.

Storage

During implementation multiple device memory options were used:

1. To operate effectively, the system requires users to enter sensitive information such as their addresses and contact numbers and store them on their mobile devices. However, these personal details could be easily detected if the mobile devices is stolen or taken from the users. Therefore, a password login is implemented. The essence of this password login is to provide authentication at a low level.
2. The application persistently stores data in order to avoid re-entering user data.
3. Before the final report is fully compiled and sent as output, it is saved in application memory. The final report output is usually in a compiled XML format file that contains all the fields of the final full report.

3.4.1 Implementation Structure

For ease of implementation on Android Integrated Development Environment, the program was structured into an activity as illustrated in Figure 3.6. An activity can be defined as a component of an application that provides a screen via which users can interact in order to accomplish a task [61]. A new activity can be referenced from the current activity using intents.

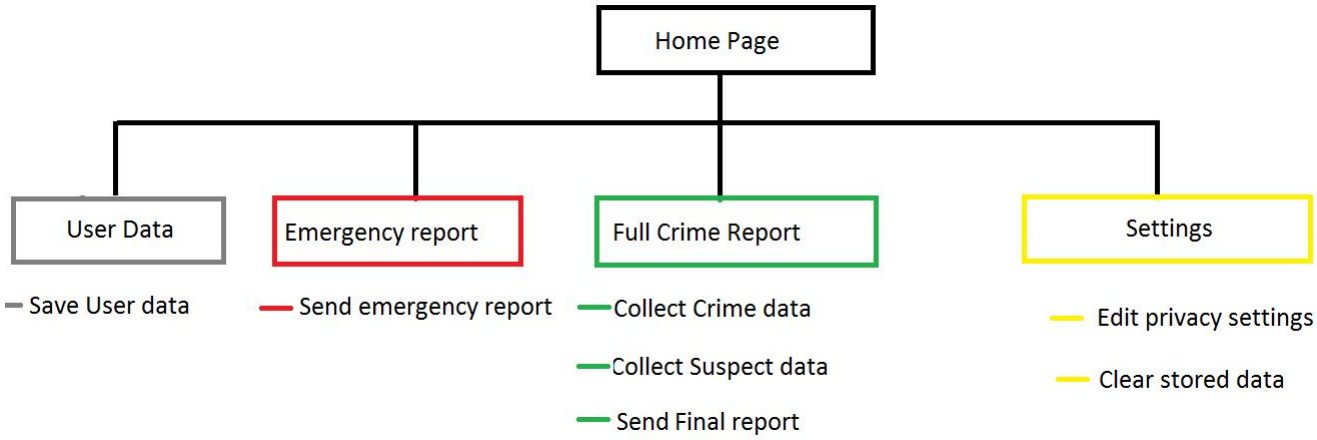


Figure 3.6: Depiction of the Activity (System Diagram) of CryHelp App

Home Page

This is the home screen of the application. Whenever the application is launched, the home page loads any incomplete crime report that the user has previously worked on and has not sent. However, if the application is launched for the first time by the user, then the User Details page will be loaded. The home page is shown in Figure 3.7 below.

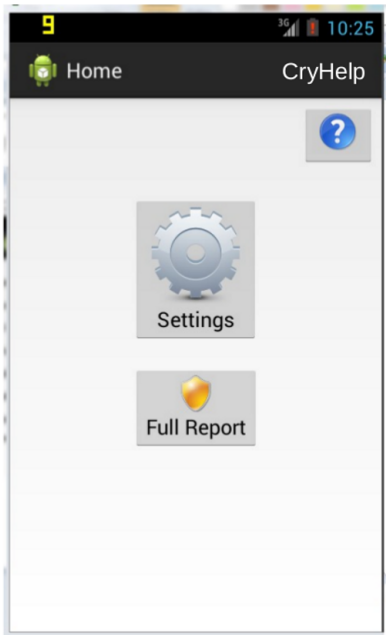


Figure 3.7: Application Main Screen

User Data Page

This page, as shown in Figure 3.8, captures the user's personal details. In addition, it enables users to set their privacy preference. The save button enables the user's details to be saved permanently so that even when the user is using the application at another time he/she will not be prompted again for his/her personal details. The cancel button closes the application.

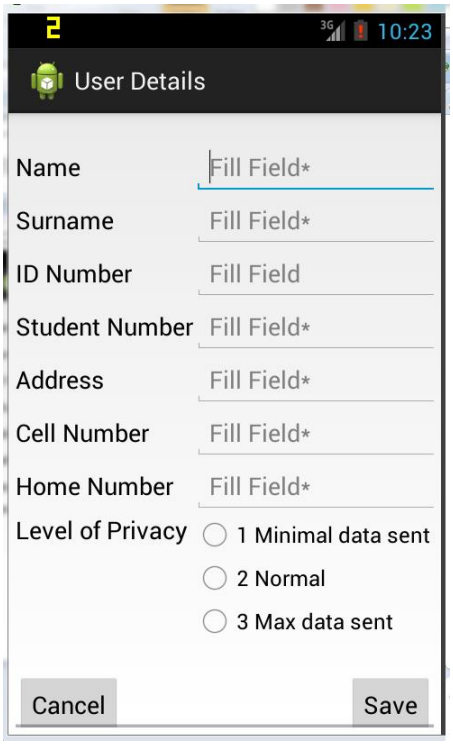


Figure 3.8: User Details Page

Full Crime Report

After a user has successfully entered and saved his or her personal details, he or she can then proceed to report the full crime. The full crime report has five pages. The first two pages focus on the crime details while pages three and four focus on the suspect details and finally page five compiles and sends the report.

- Crime Details Page

  The crime details pages shown in Figure 3.9 enable the user to provide the details of the crime.

The program has features such as a help button, which provides tips on how to navigate through the application and the camera button, which allows the user to take a picture and attach it to the report. Generally the application minimizes the time a user will spend in reporting a crime by providing tips such as a list of possible crimes, automatically detecting the user's current location, time-stamp etc.

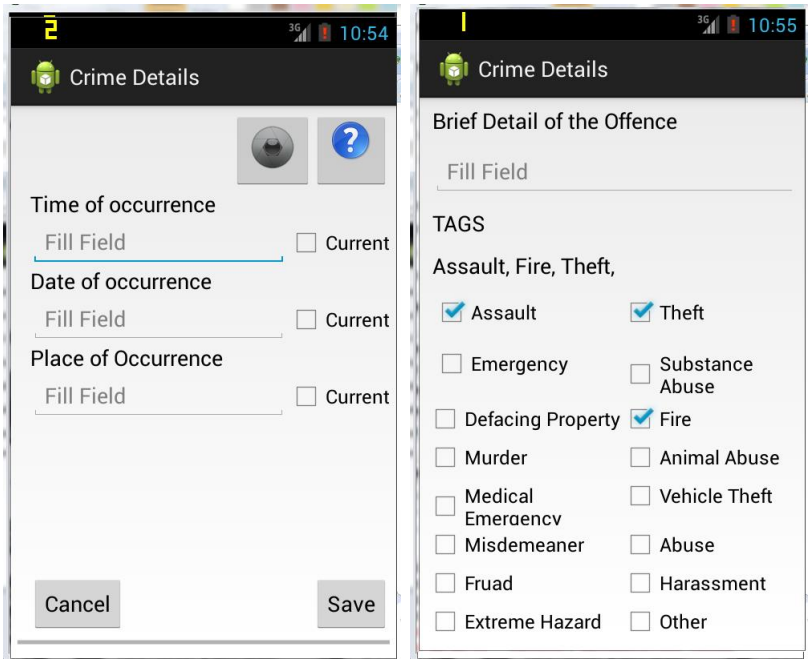


Figure 3.9: CryHelp: Crime Report Details Pages

- Suspect Details Page

  The suspect details page shown in Figure 3.10 facilitates the collection of data about the suspect in a crime. The page also has a camera feature that allows the picture of a suspect to be taken and tagged. After filling in the suspect details, the user is then given the option of sending the overall report.

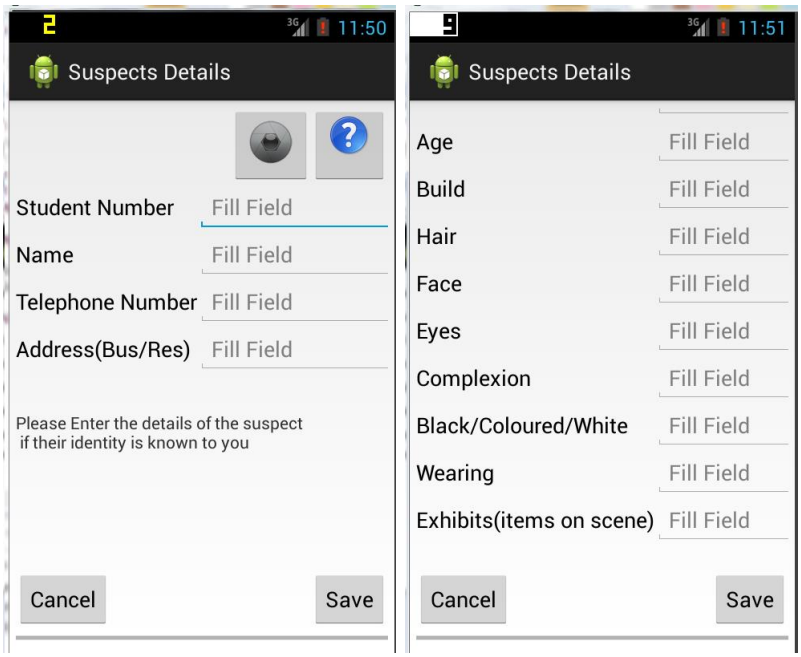


Figure 3.10: CryHelp: Suspect Details Pages

3.5 Chapter Summary

This chapter began with a presentation of a general framework for the proposed crime reporting application system. This was followed by a brief discussion of each of the components of the system, namely the: user layer, network layer, algorithmic layer and application layer was discussed. Afterwards, more details on the user layer, which was the focus of this chapter, was presented. Thereafter was a discussion of how the user layer was designed through the use of low and high fidelity prototypes as well as how these prototypes evolved through two different levels of iteration. Furthermore, an extensive discussion of the crime-reporting application, CryHelp App, which evolved as a result of the iterations was documented. The CryHelp App was used for data collection for the algorithmic layer, which is discussed extensively in subsequent chapters.

Chapter 4

Buffering Streaming Data

4.1 Introduction

This chapter re-establishes the research rationale and presents the model and algorithm specification for the adaptive buffer re-sizing scheme (ABRS). ABRS focuses on preserving privacy in a manner that minimizes information loss and delay (expired records) by taking into consideration the distribution of data in the current sliding window and future sliding window(s). To predict the distribution of data and rate of anonymization in the future sliding window(s), the Poisson probability distribution was adopted. Other possible probability models that were considered include the binomial and normal distributions. However, Poisson distribution was chosen because it focuses on finding the occurrences of an event within a specified period [6, 49, 1]. Furthermore, Poisson distribution is applicable when the possible number of events can take up whole numbers and the average frequency of occurrence for the period under consideration is known [6, 1]. Of interest in this research is finding the occurrence rate of anonymization across sliding windows. More details about how this works is documented in this chapter.

4.2 System Overview

Figure 4.1 presents an overview of a target system for data stream anonymization by revealing how different layers interrelate in the real world. Of interest is how policy makers can benefit from the output of the research without infringing on users' privacy.

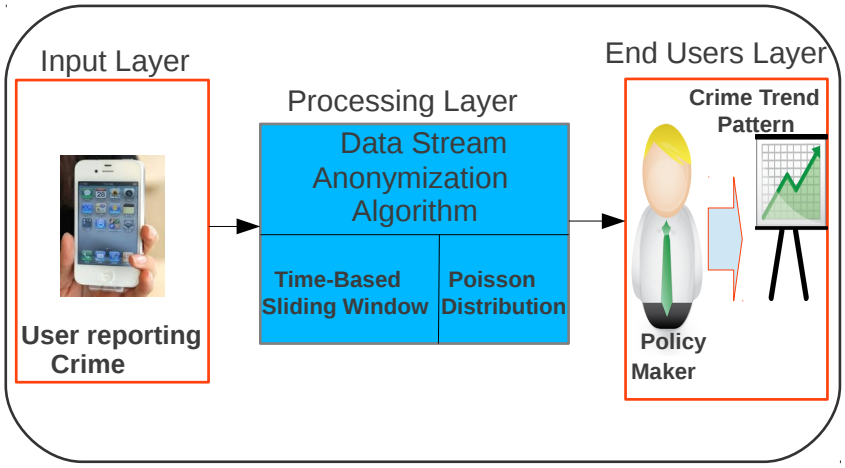


Figure 4.1: Depiction of Interaction that Takes Place in the Algorithmic Layer

4.2.1 Input Layer

Essentially this layer illustrates how data flow into the data stream anonymization framework. Typically data flow into the framework whenever a crime is being reported. Details about how crime reporting application, CryHelp, was developed have been documented in Chapter 3.

Input Definition and Formulation

Let V be a set of crime victims, where each victim, say, $v_i \in V$, is defined by a set of attributes, $A(v_i)$. Our interest lies in how the set of attributes, $A(v_i)$, can be anonymized in real time such that data utility is maximized while maintaining privacy.

To set the context for the data stream anonymization, the set of attributes, A, was divided into three subsets. That is, $a_e \subset A$, $a_q \subset A$ and $a_s \subset A$ where a_e represents explicit attributes, a_q represents quasi-attributes and a_s represents sensitive attributes. Explicit attributes are attributes that directly identify an individual e.g. a unique ID number [24]. QIs are attributes that have the potential to identify an individual, e.g. date of birth, sex. Sensitive attributes are attributes whose value must not be disclosed, e.g. crime type or disease suffered (in the case of a health scenario) [89]. In order to decide if a tuple has exceeded its time-delay constraint, additional attributes such as arrival time, expected waiting time and entry time were included.

4.2.2 Processing Layer

The processing layer, which is also referred to as the algorithmic layer, basically performs the anonymization process in real time. As stated in the introductory chapter, the concept of anonymization was basically conceived for static data. Though recently efforts have been made to incorporate anonymization into the data stream, there is still a need for more research effort, particularly in a manner that takes distribution of data streams into consideration, ultimately to reduce IL.

To adapt anonymization schemes into data stream, the concept of a time-based sliding window and Poisson distribution was introduced. The time-based sliding window is a logical window type defined by time-unit whose scope is usually defined by a function of time bounded by a lower limit and upper limit [67]. On the other hand, a count-based sliding window retains N most recent tuples, where N is user defined and an integer usually greater than zero. The advantage of a time-based sliding window over a count-based one is that it is sensitive to records that are delay-bound. Poisson distribution enhances anonymization and data utility by taking into consideration both the present stream (or data) under consideration and the future stream (data).

Typically the sliding window (also referred to as a buffer) is responsible for temporarily storing the sequence of data streams that come in from the input layer. Then, the anonymization process is applied to records in the buffer by grouping (clustering) data, using generalization and suppression in order to ensure that each group (cluster) has a size of at least k. The anonymization algorithm also ensures each item of data is placed in a group (cluster) that yields the lowest IL. The anonymization scheme also applies ℓ-diversity and t-closeness privacy measures to address the shortcomings inherent in the basic privacy measure (i.e. k-anonymity).

For the buffer component, the general mechanism adopted in most literature is simply to assign a fixed size to the buffer limit [10, 29, 101]. However, this has the limitation of leading to high IL and expiration of records [59] [71]. Therefore, ABRS is proposed as an advancement to overcome the limitation. By adaptive, this means that the buffer size can either increase or decrease. The motivation for increasing the buffer size may be the increasing proliferation of mobile devices and the increase in the number of persons who can report crime incidents via their wireless devices. On the other hand, the motivation for decreasing the window size is based on attempt to reduce the rate of record expiration, especially when it is observed that the flow of data in the stream is low and could lead to a high rate of record expiration. Following is a presentation of some concept on streaming buffer data, as well as explanation

of ABRS.

Buffer Streaming Data

Three major factors affect the privacy level that can be achieved and the rate at which information is lost during data stream anonymization. These factors are:

1. Buffer size
2. Arrival rate of streaming data
3. Rate of suppressed (unanonymizable) records

The buffer size, which could either be time-bound or tuple-bound influences the amount of information that can be held or stored. By time-bound, it means that the buffer can exist for a period T while by tuple-bound it means that the buffer can contain μ records. Former approaches that use a static means of buffer size tends to either lose information or adversely affect the privacy level obtained when there is intermittent flow of data in the stream. Thus, employing an adaptive mechanism is a useful concept that helps to resolve this problem. The adaptive mechanism dynamically adjusts the size of the buffer as streaming data flow in and also allows overlapping of sliding windows.

Figure 4.2 presents an overview of how buffer size can be dynamically adjusted as streaming data arrive. A Data Streams, DS, is defined as a real-time and continuous data flow ordered implicitly by arrival time or explicitly by timestamps.

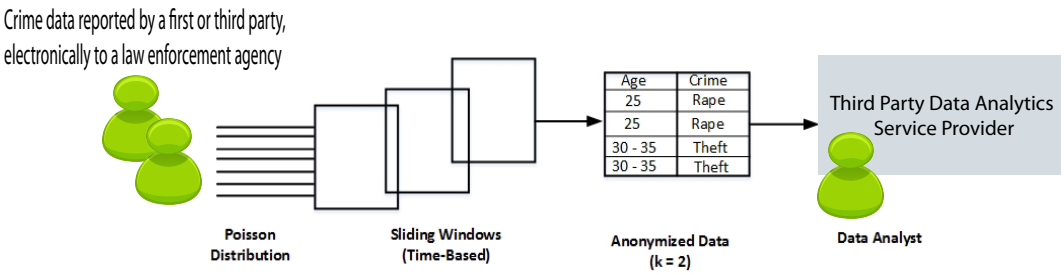


Figure 4.2: Overview of Buffer Resizing Process

In order to guide the concept of the buffer streaming data, the following definition follows:

Definition 1:

(Sliding Window (sw_i)) A sliding window, say sw_i, is a subset of the data stream, DS where DS = {sw$_1$, sw$_2$, sw$_3$,..., sw$_m$} implies that DS consists of a set of m sliding windows.

The sliding windows obey a total ordering such that for every $i < j$, sw_i precedes sw_j. Each sliding window, sw_i which is $sw_i \subset DS$, only exists for a specific period of time T and consists of n finite and a varying number of records (R), such that sw$_i = R_0, ..., R_{n-1}$.

Since there are varying records, the ability to predict the distribution of incoming stream for a particular period based on past stream behavior is useful for adjusting the buffer size. For this purpose, the Poisson probability distribution model was used to predict the rate of data flow in the next sliding window, sw$_{i+1}$, based on the rate of flow in a previous sliding window, sw$_i$. The Poisson model was used because the Poisson distribution is concerned with the number of successful predictions that an event would occur in a given unit of time. This property of the Poisson distribution makes it possible to view the arrival rate of the reported crime data as a series of events occurring within a fixed time interval at an average rate that is independent of the time of occurrence of the last event [6, 49, 1]. In the model, only one parameter needs to be known: the rate at which the events occur, which in this case is the rate at which crime reporting occurs.

4.2.3 Adaptive Buffer Re-Sizing Scheme

To achieve the objective of carrying out data stream anonymization in a manner that takes the distribution of the present data stream and future data stream into consideration, a model called ABRS, which combines concepts from the time-based sliding window and Poisson distribution, is proposed. Figure 4.3 outlines the six main phases involved in ABRS. Details and supporting examples about these phases follow.

Phase 1: Initial Buffer Size

The size of the buffer is first set to some initial threshold value, T. Let T be the time for which a sliding window, sw$_i$, exists, where T is a time value that is bound by a lower bound value, t_l, and an upper bound value, t_u. For example, in previous work [101], values between 2000 ms and 5000 ms have been used as the time interval in which a record can stay in the buffer. In line with this, the threshold values

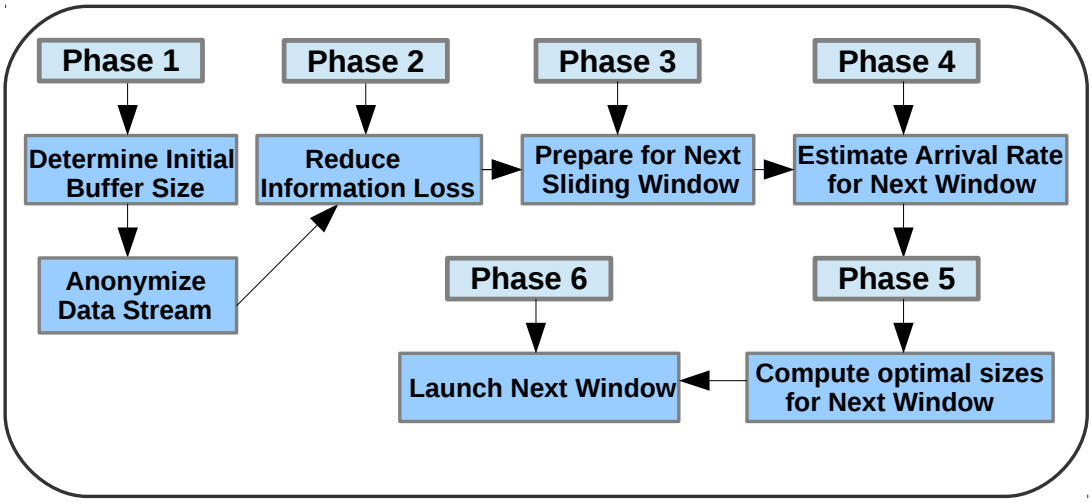


Figure 4.3: Phases of Adaptive Buffer Re-sizing Scheme

are, t_l = 2000 ms and t_u = 5000 ms. Afterwards, the following occurs:

1 The anonymization algorithm is applied to the data that were collected in the sliding window, sw_i, during the period T. Details of the anonymization algorithm follows in section 4.2.6.

2 Essentially $\text{sw}_i = T$.

3 Ideally, all records that are not anonymizable from the data collected in sw_i are either suppressed or excluded from the dataset released for publication. An effect of this is that it leads to an increase in IL and offers low data utility. Therefore, there is a need to attempt to reduce IL. Consequently, Phase 2 is born.

Example 4.1: In order to understand how phase 1 works, consider the dataset provided in Table 4.1 that has a time-defined size of 5000 ms for a sliding window, sw_i. This implies that the anonymization algorithm is applied to the data that were collected in the sliding window, sw_i, during the period $T = 5000$ ms. The anonymization process was handled with a k-anonymity scheme in which $k = 3$ is used as the anonymization metric. k was chosen as 3 because of the small dataset, which consists of only

10 records. A higher value of k will lead to higher IL. The dataset illustrated in Table 4.1 is a collection of crime reports observed in the data stream within time T; the QIs are "age" and "address" while the sensitive attribute is "crime reported". To achieve anonymization in Table 4.1, the address taxonomy tree in Figure 4.4 was used for the attribute named residence and intervals for the attribute named age to cluster records that belong to the same parent node; this results in Table 4.2. Afterwards, ℓ-diversity and t-closeness privacy schemes were applied to mitigate the vulnerabilities inherent in k-anonymity. All records that are not anonymizable from the data collected in sw_i are either merged with other cluster(s) or suppressed (excluded) from the dataset released for publication. By unanonymizable, this means that such records belong to a cluster whose total number of records is less than k. In other words, such records could also be seen as outliers. An effect of this is that IL increases and data utility declines. In an attempt to reduce IL caused by either merging or suppression, phase 2 is born.

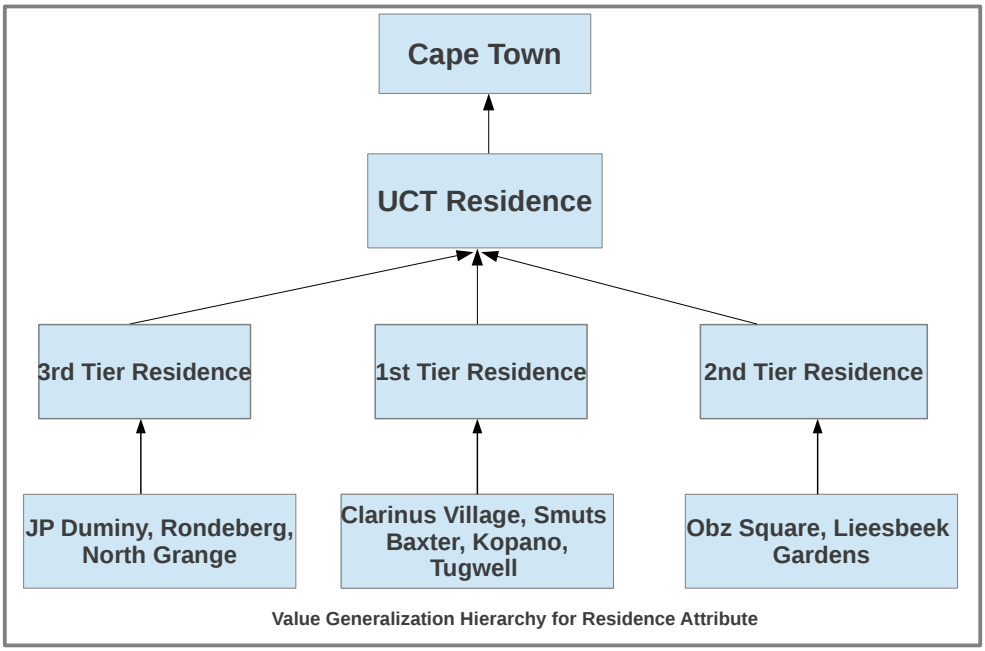


Figure 4.4: Residence Taxonomy Tree

Table 4.1: Data for Sliding Window, $T = \text{sw}_1 = 5000$ ms; $T_A = 80$ ms

Record ID	**Reported Crime**	**Age**	**Address**	**WaitingTime** $= T_S$
1	Vandalism	60	Room 3.4 Clarinus Village	4782
2	Murder	20	Flat 21 J.P. Duminy Court	4017
3	Theft	50	Room 3 Flat 2 Baxter Hall	3361
4	Corruption	60	Room 210 Kopano	2566
5	Rape	30	Room D Flat 603 Rondeberg	2118
6	Burglary	70	Room 102 Tugwell Hall	2069
7	Forgery	35	Room C Flat 207 Liesbeeck Gardens	1492
8	Arson	40	Room 110 Smuts Hall	1214
9	Drunken Driving	50	Room 6001 Obz Square	417
10	Robbery	40	North Grange House, Broad Road	100

Table 4.2: Results for k-anonymization of $\text{sw}_1, k = 3$ and $T = 5000$ ms

Cluster 1	**Cluster 2**	**Cluster 3**
(3rd Tier Residence, 20-40)	(2nd Tier Residence, 35 - 50)	(1st Tier Residence, 40-70)
2	7	1
5	9	3
10		4
		6
		8

Phase 2: Reduction of Information Loss

This second phase attempts to reduce IL that is likely to occur as a result of suppressed/unanonymizable records in Phase 1.

Let C be a set of anonymized clusters where C = $\{c_1, c_2, c_3, ..., c_m\}$. A cluster is anonymized if it satisfies the k-anonymity requirements. The k-anonymization algorithm requires that records be classified into clusters of at least size k, such that each record in the cluster is indistinguishable from at least *k-1* records. A record, R_i, is unanonymizable or suppressible if it does not fit into any of the clusters in set C in a manner that either maintains or reduces its IL. In other words, such records could also be seen as outliers. An effect of this is that IL increases and data utility declines. Therefore it is necessary to provide a mechanism to reduce IL.

To calculate IL with respect to the number of records, i.e. deviation of anonymized data from its initial form, the formula in equation 4.1 was used, as it is in [38]. This metric was adopted because it is a benchmark in many data stream anonymization schemes [10, 29, 101].

$$\text{InfoLoss} = \frac{M_P - 1}{M - 1} ... \tag{4.1}$$

M_p is the number of leaf nodes in the subtree at node P and M is the total number of leaf nodes in the generalization tree.

In order to minimize the rate of IL due to the unanonymizable records, these unanoymizable records were either included in a subsequent sliding window, say sw_{i+1}, or incorporated into already anonymized clusters (reusable cluster) of data that are similar in terms of content. Furthermore, a reusable anonymity cluster is described as one that has successfully published a set of anonymized records whose privacy and IL levels are not negatively affected by the inclusion of the suppressed record(s).

Example 4.2: To understand Phase 2, the output of the k-anonymization process in Phase 1, i.e. Table 4.2, is searched for unanonymizable/suppressed records. From Table 4.2 it is noted that records with ID 7 and 9, i.e. R_7 and R_9, are not anonymizable with the dataset in the current sliding window sw_1 because the group of records they are categorized into does not contain sufficient records to meet the k-anonymity requirement of $k = 3$. Therefore, there is a need to decide whether to process the records R_7 and R_9 in the next sliding window sw_2 or whether to find an appropriate reusable cluster

into which to incorporate the records instead.

Phase 3: Inclusion of Suppressed/Unanonymizable Records in the Next Sliding Window

One of the goals of this solution is to reduce the number of expired tuples. A tuple is considered as "expired" when it remains in the system for longer than a pre-specified threshold called delay [101].

Therefore, in order to determine whether or not a suppressed record, SR_i, can be included in a subsequent sliding window, say sw_{i+1}, its expiry time T_E is computed using equation 4.2. Afterwards, the value of T_E is compared to the bounds for acceptable sliding window sizes $[t_l, t_u]$. Thus, the expiry time of SR_i, is computed as follows:

$$T_E = \text{sw}_i - T_S - T_A \quad ... \tag{4.2}$$

where sw_i is the time-size of the previous sliding window, T_S is the time for which a suppressed record, SR_i was stored in a previous sliding window, sw_i, and T_A is the time it took to carry out anonymization in the previous window, sw_i.

Example 4.3: Following from Table 4.2, records R_7 and R_9 are unanonymizable because they are categorized into a group that does not contain sufficient records to meet the k-anonymity requirement. That is, we could say they are outliers with respect to the current distribution of data in the present stream. Therefore it is necessary to predict if a future data stream will have similar records in order to provide better anonymization and improve data utility.

In order to determine whether or not to include these records in the next sliding window, sw_2, the remaining time $T_E(R_i)$ of both records are computed and both values compared to the bounds for acceptable sliding window sizes. From Table 4.1, $T_S = 5000$ and $T_A = 80$. Therefore, $T_E(R_i)$ is computed using equation 4.2 by subtracting T_S and T_A from $\text{sw}_1 = T$, which in this case gives $T_E(R_7) = \text{sw}_1 - T_{S_7} - T_A = 5000 - 1492 - 80 = 3428$ ms and $T_E(R_9) = \text{sw}_1 - T_{S_9} - T_A = 5000 - 417 - 80 = 4503$ms. Given that $t_l = 2000$ ms and $t_u = 5000$ ms, it follows that $t_l \leq T(R_7), T(R_9) \leq t_l$ and it can be concluded that it makes sense to examine further if R_7 and R_9 should be incorporated into the sliding window sw_2.

In cases where such suppressed records cannot be considered for the next sliding window, say sw_{i+1}, the concept of a reusable anonymity cluster as discussed in phase 2 is used. Once again,a reusable anonymity cluster is described as one that has successfully published a set of anonymized records whose privacy and IL levels are not affected negatively by the inclusion of the suppressed record(s).

Phase 4.4: Determination of Arrival Rate

This phase further examines suppressed records for inclusion in the next sliding window, say sw_{i+1}. The main drive behind this phase is to predict the similarity between the data stream distribution in the current sliding window and the next sliding window.

Let U be a set of unanonymized clusters of an anonymization process where U = $\{u_1, u_2, u_3, ..., u_n\}$. A cluster is unanonymized if it does not satisfy the k-anonymity requirement. The requirement for k-anonymity is that a cluster contains a minimum of k records where $k \geq 1$.

Starting with the unanonymizable cluster that has the suppressed record, SR_i, with the lowest T_E and whose value falls within the acceptable sliding window bound, $[t_l, t_u]$, the algorithm checks for other suppressed records that belong to the same unanonymized cluster, u_i, as SR_i. After which it proceeds to find the rate of arrival, λ, of data in that unanonymized cluster u_i, within the time interval, sw_i and compute the expected arrival rate of records required to anonymize SR_i within its expiry time, T_E using equation 4.3.

$$\lambda = \frac{+|u_i|}{sw_i} \times T_E \quad ... \tag{4.3}$$

Example 4.4: Building on example 3, to decide what the optimum size of sw_2 should be set to, the expiry time, T_E, of the suppressed records in sw_1 is taken into consideration. Since $T_E = 3428$ ms for R_7 and 4508 ms for R_9, $sw_1 = 5000$ ms and $k = 3$ is being used as the k-anonymization metric and both records (R_7 and R_9) fall under the generalization attributes of crime = "2nd Tier Residence" and age = "35 - 50" , it is required that at least one similar record arrive during sw_2 in order to ensure that anonymization succeeds, thus avoiding IL from record expiry due to failure to anonymize the records. Starting with the least T_E, 3428, $\lambda_{i+1} = \lambda_2$ for R_7 is computed as follows:

$$\lambda_2 = \frac{\text{Number of Records}}{sw_1} \times T_E = \frac{2}{5000} \times 3428$$

R_7 gives $\lambda_2 = 1.37$.

Phase 5: Optimal Size for the Next Sliding Window using Poisson Probability

Let λ be the expected arrival rate of data in an unanonymized cluster, u_i, in a sliding window, sw_i and *n* be the number of records u_i required to undergo proper anonymization. Then, the probability that an unanonymizable/suppressed record SR_i in u_i would be anonymized in the next sliding window, sw_{i+1}, can be calculated using equation 4.4,

$$f\left(\text{sw}_{i+1}, \lambda\right) = \text{Pr}\left(\text{i} = 0 \ldots. \text{ n}\right) = \frac{\lambda^i e^{-\lambda}}{\text{i!}} \quad \ldots \tag{4.4}$$

where λ is the expected data arrival rate, e is the base of the natural logarithm (i.e. $e = 2.71828$), *n* is the total number of observations and *i* is the number of records under observation. Therefore the probability of having *n* or more than *n* records arriving in the stream within time T_E is

$$1 - \sum_{i=0}^{n-1} Pr \quad \ldots \tag{4.5}$$

where *Pr* is the probability outcome of equation 4.4.

The expected arrival rate, λ, from phase 4 is then used to determine the probability of arrival of the minimal number of records, *n*, which is required in order to guarantee that delaying the anonymization of the suppressed record, R_i, to the sliding window sw_{i+1} will not adversely increase IL. This is achieved by finding the probability that *n* records will actually arrive in the data stream within time, T_E, in order to anonymize the suppressed record, R_i. The expression in equation 4.4 is used to compute the probability of having $i = 0 \ldots n$ records arrive in the stream within the period T_E and equation 4.5 to determine the probability that *n* or more than *n* records will arrive in the stream within T_E.

Example 4.5: From example 4, the number of suppressed records in the unanonymizable cluster ("2nd Tier Accommodation", "35 - 50") is two i.e. R_7 and R_9. Substituting $\lambda_2 = 1.37$ into equation 4.4 and subsequently into equation 4.5, we find the probability Pr ($\geq$ 1 record belonging to group 2 arrive in the next 3428 seconds) = 1 - Pr (0) = 1 - 0.25 = 0.75.

Phase 6: Final Decision on the Size of the Next Sliding Window

Let δ be a pre-set probability threshold and *Pr* be the result of equation 4.5. If $Pr \geq \delta$ then the size of the next sliding window, sw_{i+1}, is set to the expiry time of the suppressed record under consideration in equation 4.4.

If the result of equation 4.4 from Phase 5 is greater than a pre-set probability threshold, δ, the size of the subsequent sliding window, sw_{i+1}, is set to the expiry time of the suppressed record under consideration. Afterwards, the suppressed record for inclusion in sw_{i+1} is then marked along with other suppressed records that have their T_E within bounds for acceptable sliding window sizes $[t_l, t_u]$. If the probability is less than the pre-set probability threshold, δ, the anonymization of the suppressed records is carried out by using a reusable cluster and calculate the size of sw_{i+1} using the next suppressed record whose T_E lies within the bounds $[t_l, t_u]$. In the event that the probability of all suppressed records is less than δ, the size of sw_{i+1} is set to a random number or some initial threshold value within the time bound, $[t_l, t_u]$. Finally, in order to decide which reusable data cluster to include a suppressed record, SR_i, our model searches for the cluster that covers the record and has the least IL.

Example 4.6: The output of example 5 is 0.75. This implies that there is a high likelihood of having one or more records belonging to group 2, i.e. "2nd Tier Accommodation", "35 - 50" (where records R_7 and R_9 belong) arriving within the next 3 428 ms. Therefore the existence time (size) of the next sliding window, $\text{sw}_2 = 3428$ ms.

4.2.4 Buffer Resizing: Algorithm

From the discussions in subsection 4.2.3, the framework for the buffer re-sizing anonymization of data streams can be summarized as follows:

The procedure, Sliding Window Existence Time (SWET) has two parameters: *i* is the *i*th sliding window under consideration and *k* is the k-anonymity requirement. Step 3 determines when to launch the first sliding window, sw_i, by randomly selecting its existence time, T, within the time bound $[t_l, t_u]$ i.e. $t_l \leq T \leq t_u$. The k-anonymization algorithm is applied to the data collected in the sliding window during the period T. Step 5 calls on procedure, Reset Sliding Window Existence Time (RSWET) to determine when to launch a sliding window, sw_i, where $i \geq 2$. Step 7 computes the processing time

Algorithm 4.1 :SWET (*i*,*K*)

1: **for** each sliding window sw_i, *i*:1 ...m **do**
2: **if** $((sw_i == 1)||(SuppRec == \phi))$ **then**
3: $sw_{iExistTime} \leftarrow T$
4: **else**
5: $sw_{iExistTime} \leftarrow RSWET(T_R, T_A, i, SuppRec)$
6: **end if**
7: $T_A \leftarrow$ Anonymization Processing Time
8: $SuppRec \leftarrow$ Suppressed Records
9: $T_R \leftarrow$ Remaining Time of Suppressed Records
10: Update Reusable Cluster (RC)
11: **end for**

used for carrying out k-anonymization. Step 8 searches for unanonymizable/suppressed records sorted by their remaining time, T_R, and grouped by their unanonymized cluster. If no suppressed records exist, then randomly select existence time, T, for the next sliding window from $[t_l, t_u]$.

RSWET has four parameters: T_R, which is a set that contains the remaining time of all suppressed records, T_A, which is the time required to carry out anonymization process, *i*, which is the *i*th sliding window under consideration and SuppRec, which is a set that contains suppressed records. RSWET starts by sorting the T_R of each suppressed record in ascending order. If any suppressed records/an unanonymized cluster exists whose $T_R - T_A \leq T_l$, then the reusable cluster will be used for its anonymization. A reusable cluster is a data structure of anonymized records whose privacy and IL levels are not negatively affected by the inclusion of the suppressed record. Otherwise, start with the suppressed record/group that has the least T_R. Then find the probability, *P*, that if such record(s) is/are included in the sliding window, sw_i, under consideration, it will be successfully anonymized before it expires.

If the λ or *P* result is greater than a threshold, δ, the sliding window size will be set to $T_{R_j} - T_A$ where T_{R_j} is the remaining time of the suppressed record under consideration. Otherwise, the algorithm fetches the next suppressed records. In the event that the value of λ or *P* for all suppressed records under consideration is less than the threshold, δ, the algorithm randomly selects its existence time, T, within the time bound $[t_l, t_u]$ i.e. $t_l \leq T \leq t_u$.

Algorithm 4.2 :RSWET($T_R, T_A, i, SuppRec$)

Sort: Sort T_R in ascending order and group by unanonymizable cluster
for j:1 ...$|SuppRec|$ **do**
  if T_{R_j} - $T_A < T_l$ **then**
    Anonymize $SuppRec_j$ using RC
    Delete $SuppRec_j$
  else
    Calculate arrival rate, λ, of $SuppRec_j$ in the sliding window, sw_i
    Find the Probability, P, of successful anonymization in sw_i
  end if
  if P or $\lambda > \delta$ **then**
    $ExistTime_i \leftarrow T_{R_j} - T_A$
    Add $SuppRec$ to sw_i
    break
  else
    anonymize $SuppRec_j$ using RC
    delete $SuppRec_j$ from $SuppRec$
  end if
end for
if P or λ for all suppressed records $< \delta$ **then**
  $ExistTime_i \leftarrow T$
end if
return ExistTime$_i$

4.2.5 Non-Poisson Implementation

In order to make a comparison between the use of the Poisson model and a non-poisson model, two existing mechanisms, namely passive-FAANST and proactive-FAANST [101], were implemented. The proactive mechanism predicts if a tuple not anonymizable in a current sliding window or round can be transfered or considered for the next round. It achieves this by using a simple heuristic (i.e. current Time - tuple Arrival Time + tuple processing Time $\geq$ delay). The passive-FAANST only checks if an unanonymizable tuple has passed its deadline or not. It achieves this by using a simple heuristic

(i.e. Current Time - tuple Arrival Time $\geq$ delay). The main difference between these delay mechanism variants and that of this model is that the model not only checks if a tuple will expire or has expired, but goes further to find the probability of it being anonymizable in the next round.

Another added advantage of this approach is that it makes use of a time-based tumbling sliding window as opposed to a count-based sliding window. In this way, anonymization is triggered on the time-sensitivity of records, also taking the rate at which records flow in the data-stream into consideration. This concept differs from the focus of many of the existing data stream anonymization algorithms that are on fast data stream and as a result do not take the rate of data arrival in the stream into consideration when determining an optimal buffer size. The buffer size and rate of arrival of the streaming data affect the rate of IL and the levels of privacy offered by the anonymization scheme.

4.2.6 Discussion

The anonymization algorithm used in this research follows the concept of k-anonymity and its variants (i.e. ℓ-diversity and t-closeness). The use of k-anonymity and its derivatives were chosen because of the simplicity [57] [25, 28, 60, 86, 75, 98, 103, 12], effectiveness [84, 81, 56] and high utility [19, 18] offered, especially when compared to an evolving counterpart such as differential privacy. In addition, research [19, 79] has shown that differential privacy is achieved as long as a dataset is anonymized using k-anonymity and t-closeness. Therefore this thesis focuses on the use of k-anonymity, ℓ-diversity and t-closeness to achieve anonymization. The framework for the anonymization of data streams using the concept of k-anonymity, ℓ-diversity and t-closeness can be summarized as illustrated in Algorithm 4.3 to 4.6.

The k-anonymity algorithm, k-anonymiser, in Algorithm 4.3 takes in four parameters, namely: *dataset*, which is the set of records under consideration for the anonymization process, *GH*, which is the generalization or hierarchy tree used for the anonymization process, *k*, which is the k-value or privacy level used for anonymization and *MaxSupp*, which is the maximum amount of suppression allowed for a dataset. The algorithm starts the anonymization process by traversing the generalization hierarchy from the leaf node to the root node. Starting from the leaf node, the k-anonymiser algorithm checks to see if records can be clustered or grouped based on their syntactic similarity such that the k-anonymity requirement is met (i.e. each group or cluster contains at least *k* records). If the k-anonymity requirement cannot be met at the leaf node, the algorithm will keep moving a level upward on the generalization hierarchy

Algorithm 4.3 :k-anonymiser(*dataset,k,MaxSupp,GH*)

if $|dataset| < k$ **then**
  Anonymization is not possible
end if
for i:1 ...$tree_{height}$ **do**
  $tree_{height}$ is a variable that stores the height of the Generalization Hierarchy, GH
  Form Equivalence Class, ECx, x:1...y, by grouping similar records using the
  same level or node on the GH
  for $EC i$ m:1 ...n **do**
    if $|EC| < k \,\&\, TotalSuppRec < MaxSupp$ **then**
      suppress all records in EC
    else if $|EC| < k \,\&\, TotalSuppRec > MaxSupp$ **then**
      i++
      attempt anonymization using the upper level or node
    else
      all EC is k-anonymized
    end if
  end for
end for

until the k-anonymity requirement is met.

The distinct ℓ-diversity algorithm in Algorithm 4.4 works a step further than the k-anonymity algorithm. After the k-anonymity process, the ℓ-diversity algorithm obtains a distribution of the sensitive values in each equivalence class. It then checks if ℓ-diversity has been achieved up to at least β degree. If the check is true, then the dataset can be said to be k-anonymized and ℓ-diversed to at least β degree. Otherwise, k-anonymization and then $\ell-$diversity take place using the next level on the tree.

The probabilistic ℓ-diversity algorithm in Algorithm 4.5 works in similar fashion to distinct ℓ-diversity. The main difference is that probabilistic ℓ-diversity ensures that in each equivalence class, the sensitive values are not greater than $1 \div \ell$.

The t-closeness algorithm in Algorithm 4.6 first of all group records into different equivalence classes. Afterwards it determines whether the distance between the probabilistic distribution of sensitive values

Algorithm 4.4 :distinct-ldiverse(*GenStep,l*)

1: ℓ is the privacy value required for $\ell-$diversity
2: Gen_Step represents the level or node on GH at which k-anonymity was achieved
3: **if** $|dataset| < k$ **then**
4: K-anonymization and ℓ-diversity is not possible
5: **end if**
6: Obtain the distribution of sensitive values for each
7: equivalence class,EC. EC is obtained from the k-anonymity result
8: **for** EC_i: i $= 1$ ...n **do**
9: Check how many of the EC has at least ℓ-distinct values
10: in the sensitive attribute
11: **if** $total_{check} < \delta$ **then**
12: GenStep++
13: attempt anonymization and l-diversity using level GenStep of the tree
14: **else**
15: all EC is k-anonymized and l-diversed
16: **end if**
17: **end for**

Algorithm 4.5 : probabilistic ℓ-diverse(*dataset*,ℓ)

1: Group similar records to form different equivalence class, EC, and ensure
2: values in each sensitive attribute is not greater than $1 \div \ell$

in an equivalence class is similar to that of the whole dataset.

Algorithm 4.6 :t-closeness(*EC,P,Q,t*)

EC is equivalence class
P = $\{p_1, p_2, p_3, ..., p_n\}$is the distribution of sensitive values in each EC
Q = $\{q_1, q_2, q_3, ..., q_n\}$is the distribution of sensitive values in the whole table
for all EC_i i:1 ....n **do**
  if $D[P,Q] < t$ **then**
    t-closeness is satisfied
  else
    t-closeness is NOT satisfied
  end if
end for

4.3 Chapter Summary

This chapter began with a discussion on the algorithmic layer. Afterwards, some basic terms such as data stream and sliding window were defined, as well as a detailed discussion on the the adaptive buffer scheme, which is divided into six phases. Each of these phases is discussed in detail, with supporting examples. In addition, algorithm structures on which these phases are based were presented. Afterwards, how the use of a time-based sliding window and Poisson model helps to achieve efficiency in anonymization when compared to some existing heuristics were highlighted. On a final note, the implementation of the concepts of k-anonymity, ℓ-diversity and t-closeness were discussed.

Chapter 5

User Privacy Preferences

5.1 Overview

In the previous chapter, the methodology on how an adaptive buffering mechanism was developed to mitigate IL while still preserving privacy was presented. This was achieved via modeling data behavior as Poisson distribution following window resizing based on three factors. The anonymization algorithm focuses on the use of a generic protection metric for all individuals without catering for individuals' real need. However, in real life, study or research shows that people have varying privacy preferences, so it makes sense to incorporate individuals' preference into data anonymization. Therefore, this chapter discusses ways to integrate users' privacy preference into data anonymization.

In order to study a user's privacy preference further, a real-life survey was conducted using our domain of interest as a case study. The purpose of the survey was to understand how people would like their data to be protected before sharing these with a third party. Thereafter, multinomial regression and association rules were used to determine factors or attributes that influence individuals' privacy preferences. The choice of multinomial regression and the association rule is based on the fact that these techniques help to study relationships among variables [33]. In addition, the choice of these techniques is determined by the nature of the response variable.

5.2 Motivation

Anonymization techniques, such as k-anonymity and its variants, basically use the same privacy level (i.e. k-value or ℓ-value) for all individuals in the data set [74, 85]. The use of the same privacy level for all users is unrealistic in real life because individuals tend to have varying privacy protection requirements [25, 97]. Furthermore, the use of the same privacy preference for all users means that individual privacy needs are misrepresented. As a result, some users may be over-protected, while some others may be under-protected. The implication of this is that over-protection could lead to high loss of information, while under-protection could lead to inadequate protection [97]. To understand how under-protection and over-protection can occur, let us assume that "Jane", who is a victim of rape, does not want an adversary to learn with high confidence that "she was once raped"; then anonymization must take place in such a manner that Jane's record is placed in a cluster whose total number of records is far greater than k to guarantee maximal protection. On the other hand, "John", whose mobile phone has just been stolen, might feel the gravity of the crime is low and might not mind his true identity being released. In this case it is not compulsory that John's record be placed in a cluster that has a high number of k records, perhaps it might not even be necessary to generalize his record. Using this illustration, John will be over-protected if his record is maximally anonymized and likewise Jane will be under-protected if her record is minimally anonymized. Since IL is a function of anonymized data, it thus implies that if a record is maximally protected against the user's preference then his/her anonymized data will cause greater loss of information. This illustration is backed up by some facts gathered during a real-life interview with crime victims. One of the discoveries is that people note that different crimes vary in severity and as a result they would like their data to be protected based on the severity of the crime. Therefore, it is necessary that anonymization takes place using people's preference to avoid being either over-protected or under-protected in a manner that takes users' concrete needs into consideration so that ultimately adequate protection is provided and IL is minimized.

In the next section, the approach to incorporate users' preferences into data anonymization was discussed in detail our

5.3 Three-Tiered Personalised Privacy Scheme

Figure 5.1 presents an overview of the integration of users' privacy preference, i.e. the Three-Tiered Personalised Privacy Scheme (TTPPS), into data anonymization.

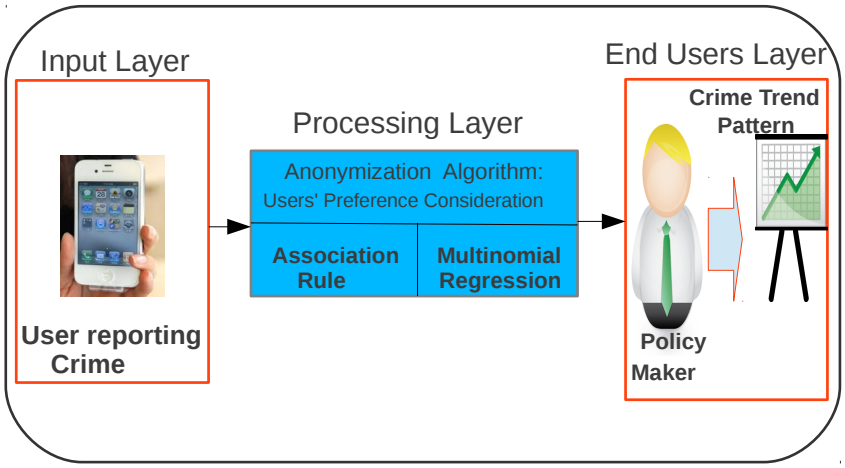


Figure 5.1: Integration of Users' Privacy Preference into Anonymization Scheme

The proposed personalised privacy scheme adopts three different levels of privacy. The basis for choosing a three-tier personalised privacy setting is that according to Chuang et al. [13] and Yuan [99], users have varying levels of privacy and can realistically choose between three levels of privacy at most. Furthermore, the effectiveness of this proposition was verified through extensive experiment. Therefore, the TTPPS is labeled as low, medium (neutral) and high. These classifications are based on the degree of sensitivity of the data that need to be anonymized.

Low Privacy Level: The privacy requirement at this level focuses on data that have a low sensitive nature. For instance, the severity of the crime of rape differs from that of the crime of theft. So theft could be considered to be of lower severity when compared to rape. Therefore a privacy-preserving scheme should take this sensitivity into consideration. Since it has been earlier established that some users want zero anonymity, it is further included that the privacy requirement at this level also represents users who have no objection to their identity being revealed. This implies that users at this level prefer high data utility to high privacy enforcement.

Medium (Neutral) Privacy Level: The privacy requirement at this level focuses on data that cannot

be classified as having either a high sensitive nature or a low sensitive nature. This level also represents users that want their data protected in such a way that information released to a third-party service provider will still be useful (i.e. low IL), while protecting some of their details. This implies that users at this level want an equal balance between data utility and privacy enforcement.

High Privacy Level: The privacy requirement at this level focuses on data that can be classified as having a highly sensitive nature. The privacy requirement at this level represents users who care a lot about their privacy. Users on this level prefer maximum protection of their data. This implies that users at this level prefer high privacy enforcement to high data utility.

Having discussed the TTPPS, which enables users to choose their personalised privacy preference, the subsequent section discusses a real-life survey that was conducted in order to see if this three-tier user privacy is usable and feasible.

5.4 Exploratory Data Analysis

Specifically, this work surveyed different groups of young people (between 21 and 40 years old) believed to have been victims of different crimes who used electronic devices. It is assumed that the willingness of an individual to share information is inversely related to his or her personal privacy preference. Therefore, the survey aims to examine the privacy preference levels (i.e. low, neutral or high) of its subjects. The justification for choosing a three-tier personalised privacy setting is that according to Chuang et al. [13] and Yuan [99], users have varying levels of privacy and can realistically choose between three levels of privacy at most. The survey also collected information on variables believed to motivate these choices. The main research question may thus be phrased as: "what are the major factors that determine an individual's privacy level preference (using the crime domain as a case study)?". Therefore, this section is a statistical analysis of the data obtained from a pilot study of the designed survey and it is hoped that it will inform interested parties how to optimize the amount of information obtainable from crime victims or potential crime victims by identifying what the significant drivers of the variation in willingness to share information are. It is acknowledged that statistical analysis of a categorical response has been widely examined in literature [3, 78].

Both visual and quantitative methods were used to summarize the contents of data collected. Such a summary is necessary to obtain preliminary information about the relationship among the variables

collected. All the collected survey data comprise 24 subjects and nine variables:

1. *Sex*
2. *Age group* (Age)
3. *Present Education Level/Occupation* (PEL)
4. *Highest Education Qualification* (HEQ)
5. *Victim of Crime* (VoC)
6. *Crime Experienced* (CEx)
7. *Preferred Privacy Level* (PPL)
8. *Share with Third Party* (STP)
9. *Reason for Choice of Privacy* (RCP)

All the subjects interviewed are postgraduate students whose *occupation* and *HEQ* are exactly related. That is, for example, a student enrolled for a PhD has a Master's degree as his or her *HEQ*. As PEL gives some clue about HEQ. Similarly, the variable, *STP*, is closely related to the variable, *PPL*. That is, a user's privacy preference dictates his/her willingness to share data with third party. Thus without loss of generality, *HEQ* and *STP* are not included in Table 5.1 and Figure 5.2 which provide summaries of the major categories of each of the variables.

Table 5.1: Description of the different categories of the surveyed variables used in the ensuing analysis. The table shows the number of subjects observed for each variable category.

Variable	*Categories*	*Subjects*
Sex	Male	18
	Female	6
Age group	26 - 30	11
	31 - 35	4
	36 - 40	9
Occupation	PhD	17
	Masters	7
Reason for privacy choice	None	5
	Adaptive	1
	Personal	8
	Insensitive	4
	Reduce crime	6
	Explicit ID delete	2

Variable	*Categories*	*Subjects*
Crime Victim?	Yes	20
	No	4
Crime experienced	Burglary	3
	Fraud	1
	Robbery	2
	Theft	12
	Mugging	1
	Assault	1
	Car snatching	1
Preferred privacy level	Low	8
	Neutral	9
	High	7

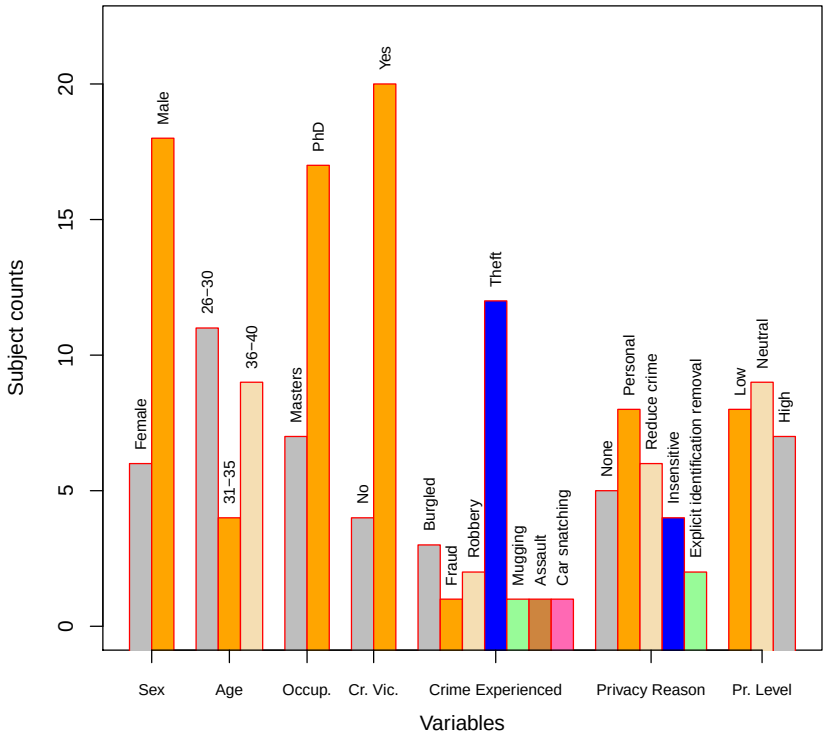


Figure 5.2: Histogram Illustrating the Distribution of Subjects Over the Different Categories of Variables Surveyed in the Primary Study of Privacy Level Preference.

5.4.1 Data Handling

Before the modelling was implemented it was deemed necessary to adjust some variables in the data set. These adjustments primarily involved collapsing redundant variable categories so that the model output could be easily interpreted. This subsection contains details of the various data adjustments that were made.

All the 24 respondents reported their PEL as either PhD or MSc, except for respondent 19 who reported Mtech. Consequently, there were two categories of PEL, namely: (i) Master's; and (ii) Doctoral. Similarly, the HEQ variable was re-classified as Bachelors, Honours and Master's.

Three respondents did not provide information about the type of crime they experienced, while the responses of the remaining 20 subjects varied among robbery, assault, burglary, car snatching, fraud, mugging, none and theft. These categories of the VoC variable were reduced to four namely; robbery, theft, other and none. There is a need to clarify the meaning of some related crimes: robbery, burglary and theft. Theft is the mildest and basic of the trio which often takes place to deprive the owner of his or her possession. Robbery on the other hand makes use of coercion to take possession belonging to another person while burglary means gaining access illegally into a building in order to commit a crime. Thereafter, only theft and robbery fore, were retained in the final classification.

The STP variable has only four records. Among those records, two are No, one is Yes and the remaining one is Yes/No. As a result, all these records were combined and labelled as answered, while the remaining 20 subjects were assigned to the category unanswered.

There was great variation among the reponses to the RCP variable. However, nine subjects attached the word "personal" to their responses, while five subjects provided no information at all. Thus, the categories adopted for the RCP variable that was included in the multinomial regression analysis presented in the next section were personal, other and unspecified.

To conclude this subsection, a summary of the adjusted variables that were used for the modeling implementation is presented in Table 5.2. All the variables are categorical. Thus, the values in the table represent the number of respondents in the corresponding categories, while the values in brackets are the proportions. For example, it is evident from Table 5.2 that out of the 24 respondents that participated in the pilot study, eight had a low privacy preference level, nine preferred a neutral privacy level and the privacy preference level for the remaining seven was high. In other words, about 33.33% of

the respondents preferred a low level of privacy, 37.50% of them preferred a neutral privacy level while about 29.17% chose a high privacy preference level. This interpretation may easily be extended to the reported statistics for high PPL.

Table 5.2: Counts of respondents that participated in the pilot privacy preference study as a function of the different classes of the qualitative variables recorded.

PPL			Sex		Age		
Low	Neutral	High	Female	Male	26-30	31-35	36-40
9	8	7	6	18	11	9	4
(0.3333)	(0.3750)	(0.2917)	(0.2500)	(0.7500)	(0.4583)	(0.3750)	(0.1667)

PEL		VoC		CEx			
Masters	Doctoral	No	Yes	None	Robbery	Theft	Other
7	17	4	20	4	3	12	5
(0.2917)	(0.7083)	(0.1667)	(0.8333)	(0.1667)	(0.1250)	(0.5000)	(0.2083)

HEQ			STP		RCP		
Bachelors	Honours	Masters	Answer	Unanswer	Unspecified	Personal	Other
4	3	17	4	20	5	9	10
(0.1667)	(0.1250)	(0.7083)	(0.1667)	(0.8333)	(0.2083)	(0.3750)	(0.4167)

5.5 Model-fitting Approach

The response of interest in the survey is the categorical PPL variable. To model PPL, this work uses two popular techniques; multinomial regression [78] and the association rule [33]. The choice of multinomial regression and the association rule is based on the fact that these techniques help to study relationships among variables [33]. In addition, the choice of these techniques is also determined by the nature of the response variable. This section deals with the multinomial regression technique while the next section is dedicated to the association rule.

5.5.1 Multinomial Regression

A motivation for the entire privacy preference study was curiosity about the relationships between the different PPLs and each of the recorded variables, namely sex, age, PEL, HEQ, VoC, CEx, STP and RCP. These differences were investigated using the available pilot study data. Given that there are three preference levels (that is, low, neutral and high) and that the variable is qualitative, the *multinomial*

Table 5.3: Multiple regression output summary. The contents of the table include values for the *odds ratio* (O.R.), the *p-value* and the lower (Low. C.I.) and upper (Upp. C.I.) bounds of the 95% *confidence interval*.

PPL		Intercept	Sex	Age		PEL
			Male	31-35	36-40	Doctoral
Low	O.R.	0.00008	0.28315	1.293×10^{-12}	3.717×10^{-09}	1.325×10^{-10}
	p-value	0.00000	0.54036	0.00000	0.00000	0.00000
	Low. C.I.	0.00005	0.00499	3.345×10^{-13}	3.717×10^{-09}	8.246×10^{-11}
	Upp. C.I.	0.00013	16.07667	4.996×10^{-12}	3.717×10^{-09}	2.129×10^{-10}
High	O.R.	0.28890	0.63897	2.169×10^{13}	1.596×10^{12}	2.053×10^{-07}
	p-value	0.00003	0.86075	0.00000	0.00000	0.00000
	Low. C.I.	0.16163	0.00429	1.214×10^{13}	1.596×10^{12}	1.149×10^{-07}
	Upp. C.I.	0.51639	95.24715	3.878×10^{13}	1.596×10^{12}	3.670×10^{-07}

PPL		VoC	CEx		
		Yes	Robbery	Theft	Other
Low	O.R.	4.186×10^{07}	4.113×10^{19}	362.3693	2.808×10^{-15}
	p-value	0.00000	0.00000	0.00000	0.00000
	Low. C.I.	2.605×10^{07}	4.113×10^{19}	225.5450	2.808×10^{-15}
	Upp. C.I.	6.725×10^{07}	4.113×10^{19}	582.1965	2.808×10^{-15}
High	O.R.	1.24378	5.691×10^{20}	5.558×10^{-11}	3.932×10^{-11}
	p-value	0.46158	0.00000	0.00000	0.00000
	Low. C.I.	0.69586	5.691×10^{20}	5.792×10^{-12}	3.248×10^{-12}
	Upp. C.I.	2.22315	5.691×10^{20}	5.333×10^{-10}	4.761×10^{-10}

PPL		HEQ		STP	RCP	
		Honours	Masters	Unanswer	Personal	Other
Low	O.R.	7.039×10^{-19}	1.325×10^{-10}	4.946×10^{15}	2.767×10^{10}	0.01013
	p-value	0.00000	0.00000	0.00000	0.00000	0.00000
	Low. C.I.	7.039×10^{-19}	8.246×10^{-11}	3.079×10^{15}	7.160×10^{09}	0.00252
	Upp. C.I.	7.039×10^{-19}	2.129×10^{-10}	7.947×10^{15}	1.070×10^{11}	0.00407
High	O.R.	1.714×10^{11}	2.053×10^{-07}	2.212055	8.141×10^{10}	1.584×10^{-09}
	p-value	0.00000	0.00000	0.71857	0.00000	0.00000
	Low. C.I.	1.714×10^{11}	1.149×10^{-07}	0.02948	4.555×10^{10}	1.584×10^{-09}
	Upp. C.I.	1.714×10^{11}	3.670×10^{-07}	165.98760	1.455×10^{11}	1.584×10^{-09}

regression analysis technique is the appropriate model for such a modelling problem.

When a qualitative variable is used in regression models, there is always a need to adopt one of the groups of the variable as the reference group. This process ensures that the effect of each of the explanatory variable can be interpreted relative to the reference group and it also ensures that the regression modelling assumption of *singularity* of the matrix of explanatory variables is not violated. The reference category for the response PPL variable was set as neutral. Let x::y imply that the reference category for the explanatory variable x is group y. Then the reference categories used for the explanatory variables are, Sex::Female, Age::26-30, PEL::Masters, VoC::No, CEx::None, HEQ::Bachelors, STP::Answer and RCP::Unspecified. Consequently, the **Intercept** term reported in Table 5.3 corresponds to the expected relative effect, on PPL, of a *female* respondent that is between *26 and 30* years old and who is currently studying for a *Master's* degree; her highest educational qualification is a *Bachelors* degree and she had provided an *answer* about whether she permitted her data to be shared with a third party but she did not specify the reason for her PPL choice. The following can be inferred from Table 5.3:

(a.) Consider the statistics for males with respect to a high PPL. The odds ratio of 0.63897 means that, compared to females, it is about $36.103\%(= 1 - 0.63897)$ less likely for males to choose a high PPL over a low PPL. However, the associated high p-value of 0.86075 implies that there is insufficient evidence in the analysed data to support any claim that the deduced difference in preference levels between males and females is significant. In other words, it is very likely that the choice between neutral or high PPL is similar between males and females. This inferred insignificance is supported by the reported 95% confidence interval (that is, $[0.00429; 95.24715]$).

(b.) The odds ratio for the intercept term with respect to low PPL implies that, on average, females between the ages 26 and 30 with the following six additional characteristics - (i) have Bachelors degree, (ii) are currently studying for their master's degree, (iii) have never been victims of crime, (iv) did not report ever having experienced any crime, (v) provided some information on whether they would permit third-party access to their data and (vi) did not justify their privacy preference choice - are about $99.992\%(= 1 - 0.00008)$ less likely to choose a low PPL over a neutral PPL. The extremely small associated p-value (that is, ≈ 0.00000) shows that the inferred relative PPL choice for the category of respondents described is quite evident. Based on the corresponding 95% confidence interval (that is, $[0.00005; 0.00013]$), the true difference in how much this category of females prefer neutral over low PPL is expected to be between $99.987\%(= 1 - 0.00013)$ and $99.995\%(= 1 - 0.00005)$.

(c.) With respect to the age variable, it can be deduced from the table that it is extremely likely

for a respondent who is between 31-35 years old to choose neutral over low PPL compared to a respondent who is between 26 and 30 years old. This claim can be justified by the extremely small odds ratio ($= 1.293 \times 10^{-12}$) that was associated with age class 31-35 under the low PPL response. Given the very small p-value (that is, 0.00000) related to this inference, it is very unlikely that the estimated odds ratio only occured by chance. The 95% confidence interval shows that, in fact, the true odds in favour of a respondent between 31 and 35 years old choosing low over neutral PPL relative to a respondent between 26 and 30 years old making a similar PPL choice is expected to be between 3.345×10^{-13} and 4.996×10^{-12}.

(d.) According to the statistics for the 36-40 age category with respect to high PPL, respondents who were between 26 and 30 years old were significantly (p-value $= 0.00000$) about 1.596×10^{12} times less likely to choose high PPL, compared to older respondents between 36 and 40 years. Stunningly, the invariant confidence interval (that is $[1.596 \times 10^{12}; 1.596 \times 10^{12}]$) implies that the evidence in the analysed data suggests that it is "certain" that the true population odds value is as reported. A similar extreme degree of accuracy can be observed in the table for CEx::Robbery, CEx::Other with respect to low PPL, HEQ::Honours and RCP::Other under high PPL. Given that the pilot study data set that is being analysed is a sample from a large population, extreme estimates similar to those highlighted here are very likely to be caused by the small size of the sample.

(e.) An odds ratio of 1.325×10^{-10} is approximately 0.00%. The same analogy may be extended to odds ratio values of 8.246×10^{-11} and 2.129×10^{-10}. Therefore, based on the inferences for PEL, it can be claimed that compared to respondents who were studying for their Master's degrees, those who were doctoral candidates were extremely unlikely to prefer low PPL over neutral PPL.

(f.) Given the high p-value inferred for VoC under the high PPL category, there is insufficient evidence in the analysed data to be able to claim that whether or not a subject had previously been a crime victim affected his/her choice when deciding on choosing between a high or a neutral PPL. Observe however that very often (given the p-value ≈ 0.0000), compared to those who had never been crime victims, subjects who had previously been victims of crime are about 4.186×10^{7} times more likely to choose low over neutral PPL.

(g.) The pattern of the relationship between CEx and PPL is noteworthy. Respondents who had experienced theft, compared to those who never had such an experience, were significantly likely to choose low or high PPL over neutral PPL. Contrarily, relative to the group of subjects who had no

CEx, subjects who had witnessed other crimes such as assault significantly preferred neutral PPL over low or high PPL. It may be worth-while to investigate other privacy techniques to cater for the category of people in the "other crime experience" class.

(h.) Compared to respondents whose highest educational qualifications were Bachelors degrees, respondents with Honours HEQ tended to prefer neutral to low PPL, but they preferred high over neutral PPL.

The above explanations and interpretations were made for just a few of the values in table 5.3. The interpretations of the other values in the table are straightforward extensions. Figure 5.3 and Figure 5.4 contain graphs that illustrate the inferred relationship between each of the PPL explanatory variables. Each data point on the graph corresponds to the proportion of respondents in the associated category as a function of the different levels of privacy preference. Observe that the lines intersect in the graphs for all the variables with very small p-values in Table 5.3. At least one of the categories for three variables (that is, Sex, VoC and STP) in Table 5.3 has p-values that are greater than 40%. The lines in the graphs for those variables do not intersect. In summary, the graphs complement the inferences from Table 5.3. That is, the graph is a visual representation of Table 5.3.

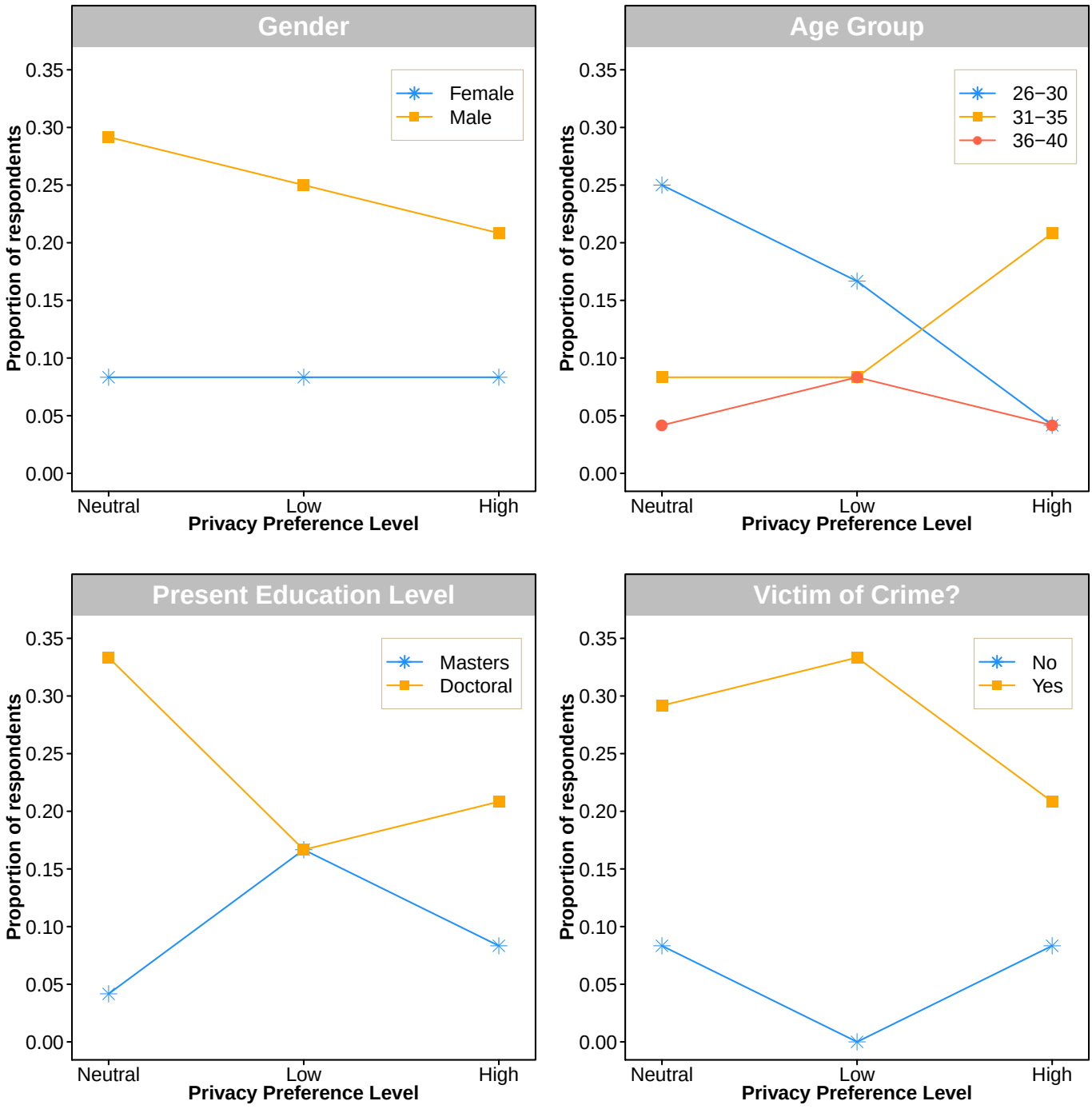


Figure 5.3: Graphs that Show the Relationships Between PPL and Each of Sex, Age, PEL and VoC.

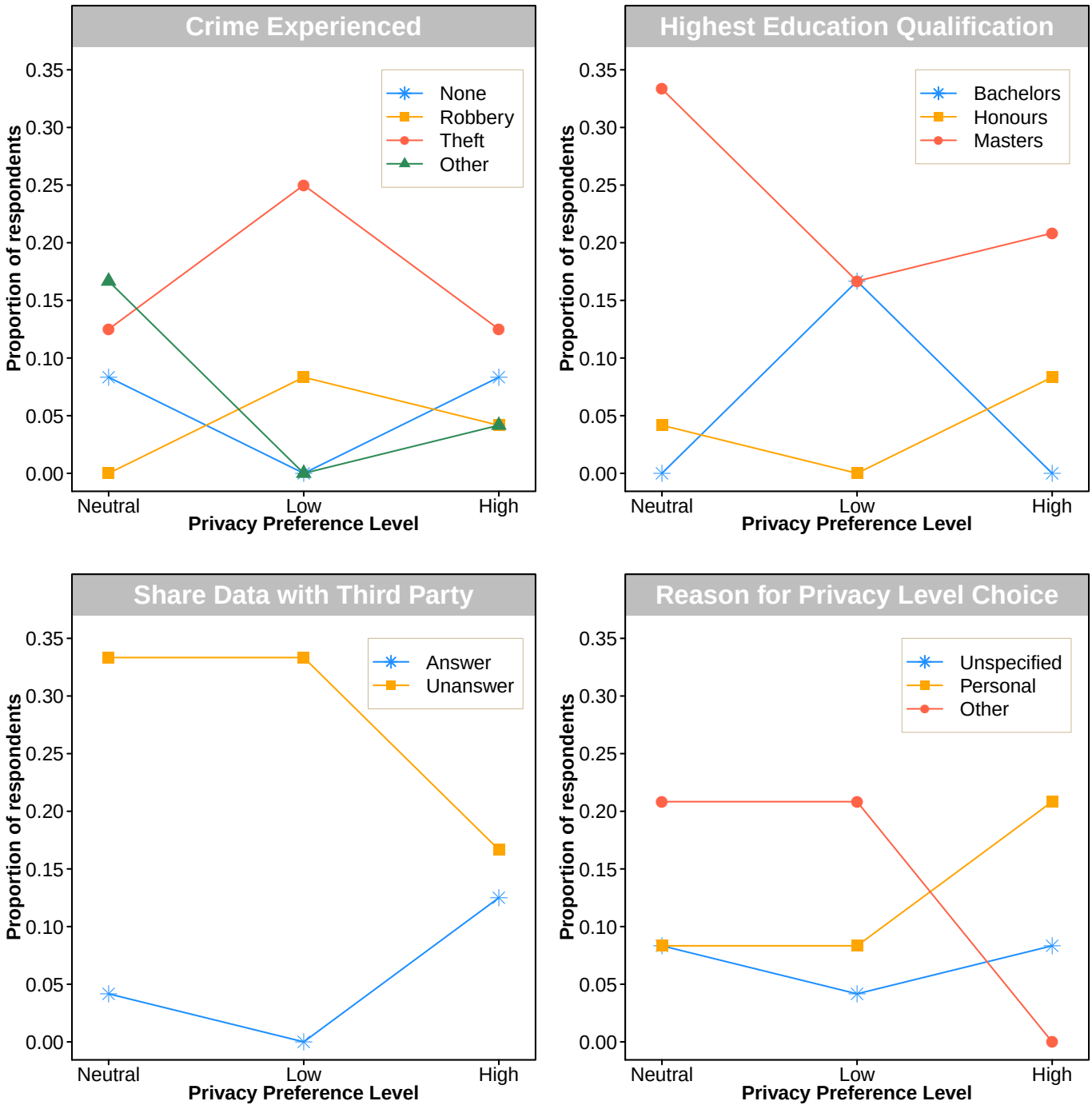


Figure 5.4: Graphs that Show the Relationships Between PPL and Each of CEx, HEQ, STP and RCP.

5.5.2 Association Rules

The association rule shows the relationship between data items. In this scenario, the use of the association rule helps to find an interesting relationship between the privacy level and users' attributes. In

other words, the association rule seeks to find factor(s) or response(s) responsible for an individual's privacy preference.

Let CV be a set of crime victims, where each victim, say, $cv_i \in$ CV, is defined by a set of attributes, A. Furthermore, $a_{ppl} \subset A$ and $a_{others} \subset A$ where a_{ppl} represent PPL and a_{others} represents quasi-attributes and sensitive attributes. Our interest lies in using the association rule to find the relationship between a_{ppl} and a_{others}. The measure used in this work identifies the relationship between a_{ppl} and a_{others} based on two important thresholds, S and $\propto$, where S (support threshold) is the frequency at which a relationship occurs in the dataset and $\propto$, which is the confidence threshold, is the ratio of the number of occurrences of such a relationship.

In summary, the subject of interest does not include all implications (association rule), but only those that are important. Here importance is measured by two thresholds introduced earlier i.e. support and confidence. Confidence measures the strength of the rule, whereas support measures how often the rule occurs in the dataset. Typically, association rules with smaller support and larger confidence should be used. Table 5.5 shows some interesting relationships that exist in our survey sample. For example, the association rule, **31 - 35** $\Rightarrow$ **High**, in Table 5.1, with a confidence value of 0.636, indicates that the rule is valid 63.6% of the time. The support value indicates the percentage of time the rule is defined throughout the dataset.

Table 5.4: Support and confidence for some association rules

Term	**Description**
D	Database
R	Record in Database
s	Support
$\propto$	Confidence
X, Y	Item Sets
X $\Rightarrow$ Y	Association Rule

Algorithm 5.1 is an a priori algorithm used for generating association rules. The algorithm takes five variables as input. The variables are D: Dataset, I: Items, L: Large Itemsets, S: Support and C: Confidence.

Algorithm 5.1 :Association Rule (D, I, L, S, C)

Input:

D: Database of transactions

I: Items

L: Large Itemsets

S: Support

C: Confidence

Output:

R: Association rules satisfying s and $\propto$

ARGen Algorithm:

R = $\emptyset$

for each l $\in$ L **do**

for each x $\subset$ l such that x $\neq \emptyset$ do

if $\frac{\text{support}(I)}{\text{support}(x)} \geq \propto$ **then**

$R = R \cup (x \Rightarrow (I - x))$

end if

Application of Association Rule

Several algorithms have been proposed to find association rules. Of these, this research employs the use of an a priori algorithm because of its ease of implementation. An a priori algorithm uses a "bottom-up" approach, where frequent subsets are extended one item at a time (i.e. candidate generation) and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

To illustrate algorithm 5.1, suppose we have a set of large itemsets:

L = ((31 - 35, High), (Female, Low), (Rape, High))

From these large item sets, L, there are three association rules. With the first association rule, (31 - 35, High), applying algorithm 5.1 to it gives us:

$$\frac{\text{support(31 - 35, High)}}{\text{support(High)}} = 63.6\%.$$

Table 5.5: Support and confidence for some association rules.

X ⇒ Y	**Support**	**Confidence**
31 - 35 ⇒ High	0.2692	0.636
Msc ⇒ High	0.2307	1
Theft ⇒ Low	0.269	0.5
26 - 30 ⇒ Neutral	0.2307	0.545
PhD ⇒ Neutral	0.3077	1
Msc ⇒ Neutral	0.3077	1.33
Male ⇒ PhD ⇒ Neutral		0.545

This means that the confidence of the association rule, X ⇒ Y, is 63.6%. Applying algorithm 5.1 to the data gathered during the crime survey, a sum of 22 rules that affect people's privacy were generated. According to the principles of the association rule, two features called support and confidence are important in choosing a valid rule. Association rules with smaller support and larger confidence should be used. Therefore, a minimum support value of 0.2 and a minimum confidence value of 0.5 were used.

5.5.3 Relationship between Association Rule and Multinomial Regression

For values of association rule confidence that are greater than or equal to 0.5 to be claimed as identical to the multinomial regression, the following is expected:

- The higher the confidence level of the association rule, the higher the corresponding odd ratio is expected to be.
- Smaller p-values are expected to be associated with higher confidence in the association rule.
- Smaller width of the 95% confidence intervals is expected to correspond to higher association rule confidence.

5.6 Integration of Three-Tiered Personalised Privacy Scheme

The integration of the proposed three-tier level privacy setting into k-anonymity starts anonymization from a general k-value into a more specific or personalised k-value. In order words, a top-down approach was adopted by moving from a generic solution into a more specific and user-centric/personalized solution. The basis for this is to ensure that under-protection does not occur.

- Level 1 Protection: Start anonymization of all tuples in the dataset using a general privacy value. At this level, the dataset is anonymized using a general privacy value. The reason for this is that according to [85], privacy is enforced if there are at least k individuals in a cluster.

  Figure 5.2 is the algorithm used to enforce privacy at level 1. The algorithm takes in two parameters, namely ds and k. ds is the set of data that needs to undergo anonymization, while k is the privacy value necessary to achieve anonymization. The algorithm starts by checking if the total number of records is greater than the privacy value, k. If it is, then the anonymization process can start using the general anonymization scheme. Otherwise, the association rule will be used.

Algorithm 5.2 :Level 1 Protection (*ds,k*)

```
1: for each record r_i, in the datastream, ds, i:1 ...m do
2:    if (|ds| < k)) then
3:       anonymization is not possible
4:       attempt anonymization using association rule
5:    else
6:       anonymization is possible
7:       attempt anonymization using a general privacy value
8:    end if
9: end for
10: return k-anonymization result
```

- Level 2 Protection: The essence of level 2 protection is to attempt to reduce overprotection that might have taken place as a result of level 1 protection. By over-protection, this means that such a record is either suppressed or placed in an equivalence class that incurs high IL.

  If such a record(s) exists, then its anonymization is attempted by using the personalised privacy preference. In other words, the three-tier model attempts to classify such a record into one of the three privacy preferences (i.e. low, medium and high) by extracting the similarity between the data and the pre-train dataset discussed earlier.

  If a record is predicted to be of low preference a minimum privacy k-value requirement of 1 will suffice for it because according to Sweeney [85], the least privacy is achieved when $k = 1$. For cases where the prediction is medium, a heuristic privacy value of $k/2$ was used because optimal privacy is achieved if the privacy value is k, so it implies that a mid-value will suffice for a medium privacy level. And lastly, if the prediction is high, the value of k is used because according to the k-anonymity concept [85], the possibility of a linking attack is reduced as long as there are at least k individuals.

  Algorithm 5.3 is the algorithm used to enforce level 2 protection. The algorithm takes in two parameters, namely ds and k. ds is the set of data that needs to undergo anonymization while k is the privacy level that is used to achieve anonymization.

Algorithm 5.3 : Level 2 Protection (*ds,k*)

1: Let $AnonResult_{unaonymized}$ be the set of records that are unanonymized or suppressed
2: **for** each record r_i, in $AnonResult_{unaonymized}$, i:1 ...m **do**
3: start anonymization using association rule
4: **end for**
5: **return** anonymization result

- Level 3 Protection: The essence of level 3 protection is to ensure that all records are adequately protected, i.e. under-protection and over-protection are minimized. It achieves this by searching for an anonymization cluster(s)/group(s) that has more than k individuals. If such a group (cluster) exists, records that can stand alone without violating the anonymity principle and the user's privacy preference are withdrawn.

  Algorithm 5.4 is the algorithm used to enforce level 3 protection. The algorithm takes in two parameters, namely ds and k. ds is the set of data that needs to undergo anonymization, while

k is the privacy level that is used to achieve anonymization.

Algorithm 5.4 : Level 3 Protection (*ds,k*)

1: **for** each anonymized cluster c_i, i:1 ...n **do**
2: **if** $(|c_i| > k)$ **then**
3: start anonymization using association rule
4: **end if**
5: **end for**
6: **return** anonymization result

5.7 Chapter Summary

This chapter began by discussing the need for users' privacy preference to be considered during the anonymization process. Then, the proposed three-tier user-defined privacy preference is discussed. Afterwards, there was a discussion of a real-life survey carried out to justify the choice of a three-tier privacy preference. This discussion brought about justification for the use of multinomial regression and the association rule to analyse the details of the survey. Finally, the algorithm structures that support a three-tier level of protection was illustrated.

Chapter 6

Implementation and Experimental Results

This chapter presents the implementation and results of the crime-reporting Application, CryApp, and the algorithms discussed in Chapters 4 and 5. In Chapter 3, a detailed framework of the system was presented, Chapter 4 discusses how the buffer is adaptively resized through the use of time-based sliding windows and Poisson probability distribution, and finally, Chapter 5 discusses how anonymization takes place through user privacy preferences. Therefore, the implementation was divided into three phases. The first phase focuses on the usability of the crime-reporting application (CryHelp), the second phase focuses on minimizing delay before anonymization through the use of an adaptive buffering mechanism and the third phase addresses privacy preservation guided by user requirements.

6.1 Experiment on Usability of CryHelp App

The experiment was open for participation to students living around the university community. The main criterion for participation was a balanced gender distribution. The age distribution of participants was between 20 and 24. It is also worth noting that all participants were familiar with the use of mobile phones for their daily activities. The participants being students of the prestigious UCT, have a good grasp of the English Language.

The experimental session for testing the usability of the crime-reporting application, CryHelp, was

divided into two main phases. The first phase focused on the the full crime report, while the second phase focused on the emergency crime report. During the first phase, participants were able to perform tasks such as: filling in personal data, suspect details, crime details, taking an image of the scene of the crime and tagging the image either suspect or victim. The second phase, which entails an emergency report of the crime, started after the first phase. Details such as personal data that were collected in the first phase were used in sending the emergency crime report.

6.1.1 Evaluation Instrument: Questionnaire

The evaluation of the software's usability was conducted through the administration of a questionnaire. The questionnaire was adopted from [48] and consists of 22 questions. Some of the key question posed centre on the following subjects:

- Simplicity of the system
- Effectiveness and efficiency of the system
- Ease of error detection
- Clarity of information on application screen
- Likability of the interface

6.1.2 Findings and Results

The focus of this section is on discussing the findings on the usability of the CryHelp App. The results of the experiments were discussed, and user experiences and feedback were reported.

1. **Ease and Time Spent on System Usage**

   The ease of completing a task and the time it took to complete the task were graded by participants. The first four questions on the administered questionnaire concerned ease of usage and time spent. Figure 6.1 shows the final average outcomes for these first four questions.

   Figure 6.1 shows that users give a similar score to the time it takes to use the CryHelp App to report a crime and the ease of using the application to report a crime. The overall average

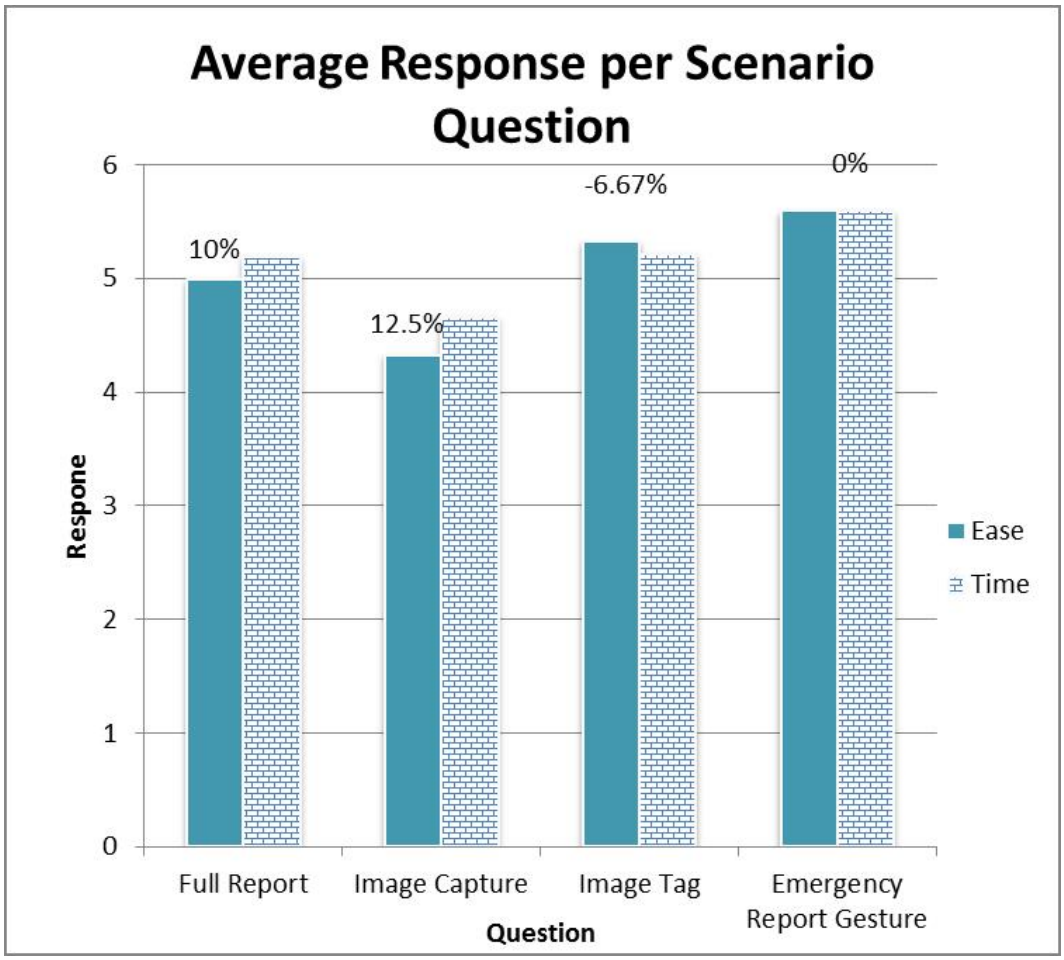


Figure 6.1: Chart Showing the Evaluation of the Ease and Time Spent Section of the Questionnaire (five-scale step), Standard Deviations of 0.54 for Ease and 0.38 for Time

difference between correlating ease of use and time taken is 3.96%. This suggests that the time it takes to perform a task is directly proportional to the perception of ease of the task. Furthermore, the result shows that the least satisfactory of all the tasks is image capture. A major factor responsible for this is that the camera button on the phone was very small and as a result could not be located easily.

2. **System Component Evaluation**

   The evaluation of the system components was based on IBM CSUQ [48]. The system components and the summary of the corresponding question(s) used for evaluation are as follows:

   – System Overall: The overall score for the system was derived from the average of answers of questions 1 to 18.

- System Usefulness: The overall score for the usefulness of the system was derived from the average of responses to questions 1 to 8.
- Information Quality: The average of responses to questions 9 to 15 was used to determine the overall score on the information provided by the system
- Interface Quality: In order to determine the quality and effectiveness of the interface, the average of answers to questions 16 to 18 were determined.

The results from these above categories play a key role in determining whether a mobile device can be used to send a crime report effectively. Figure 6.2 shows the result of each component of the system. From the figure, it can be seen that overall the system was well received, with a score of 77.06%. This implies that users found the system usable and effective for reporting a crime. It is interesting to find that the interface quality (78.33%) was the most appreciated aspect of the system. A possible factor responsible for this is that the design process was centred on the users. It can therefore be concluded that mobile applications are feasible platforms for crime reporting.

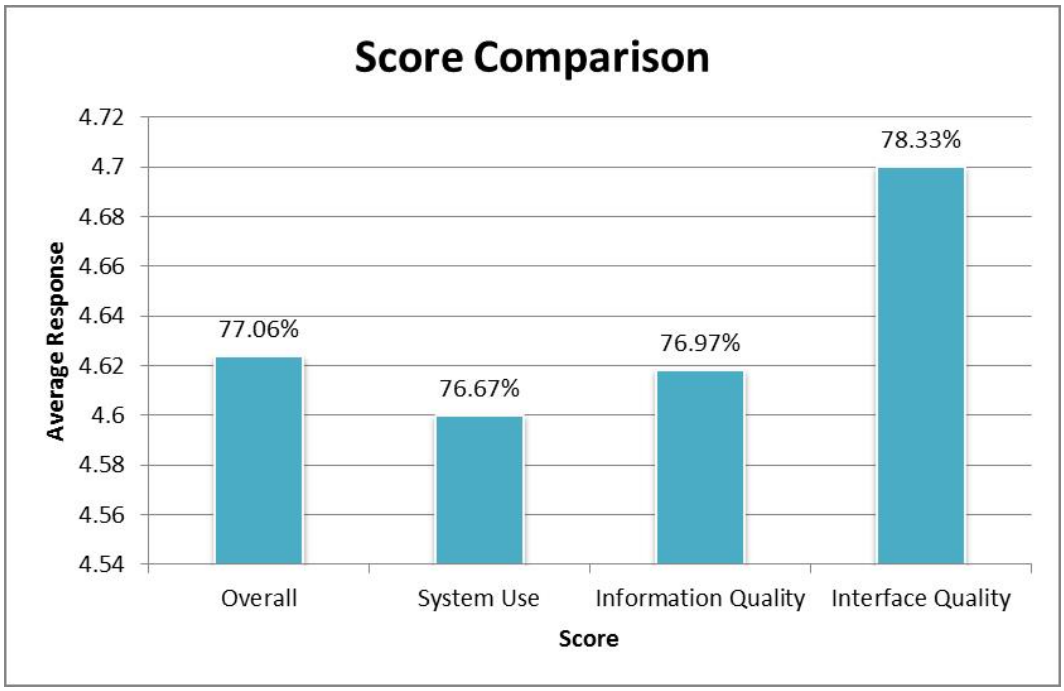


Figure 6.2: Bar-Chart Showing the Evaluation of the System Components Breakdown with Standard Deviation of 0.05

3. Time Analysis

Another important measure considered during evaluation was the time taken for users to report

a crime effectively using the CryHelp App. A summary of the time it took participants to report a crime successfully is shown in Figure 6.3. The time it took users to use the CryHelp App to report a crime successfully varied between 124.55 and 580.3 seconds, with a standard deviation of 164.18. An important reason for this large difference in time is that users were reporting different types of crime. Also, users had different privacy preferences, which reflected in the type of details they were willing to release during crime reporting. Overall, participants were satisfied with the time they spent using the CryHelp App to report crime.

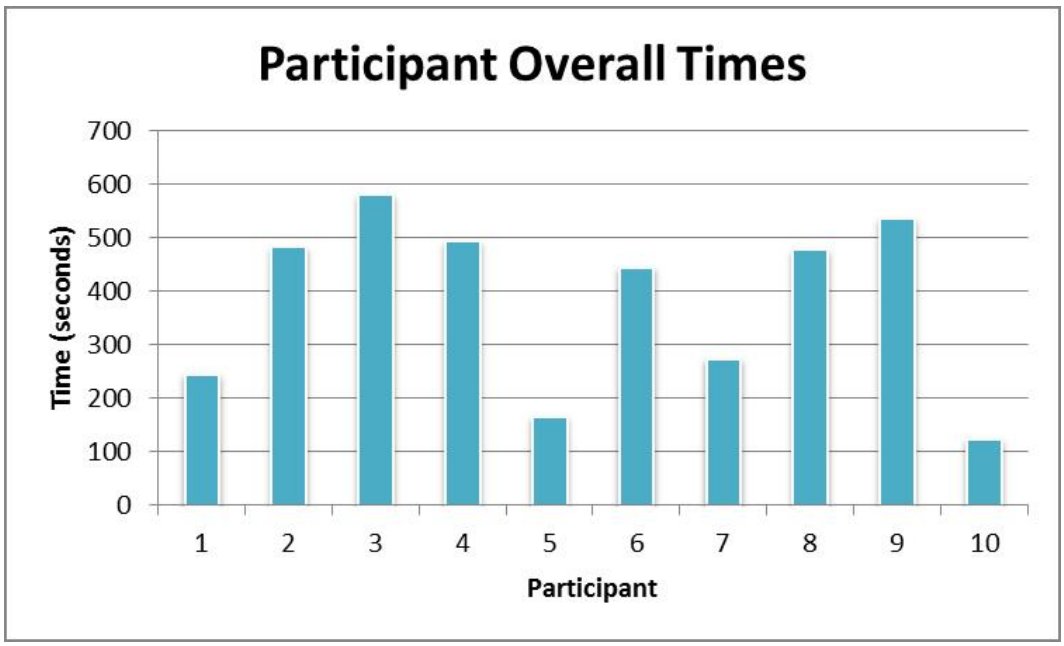


Figure 6.3: Bar-Chart Showing Time Taken to Report a Crime with Standard Deviation of 164.18

4. User Experiences and Feedback

Interaction with users showed that they were generally satisfied with the system. The users found the interface easy to use without any need for prior training, yet the users noted some key difficulties when using the application. A key one has to do with the OS (android OS) platform on which the application was running. Users who were not familiar with an android OS obviously found it somewhat challenging to use the CryHelp App. For instance, many such users found it difficult to locate the return button on the phone. When using the application, a major challenge users faced had to do with the type of values a field could take. So users needed clarity in this regard.

In this section, the experiments were discussed and an evaluation carried out on the usability of the crime-reporting application system, CryHelp. Results from our evaluation bode very well for the feasibility of a mobile solution for crime reporting. The following section offers a discussion experiments on anonymization carried out in real time during crime reporting.

6.2 Experiments on Anonymization

Having established that the CryHelp App provides a platform for reporting crime in an efficient and secure manner, experiments were carried out on the anonymization of crime records in real time. The proposed framework as explained in Chapter 4 was implemented in Java. Testing experiments were set up to measure performance, usability and security. Performance was measured in terms of turnover time for the running of the algorithms to ensure that a dataset was anonymous. For usability, IL serves as a gauging measure. Security was measured by determining if the minimum security threshold had been reached for all tuples in a data set. The experiments were conducted on an Intel Core i5-3210 2.50 GHz machine with 4GB of random access memory. The operating system used was Ubuntu 12.10. Concepts from the CSE 467 anonymization toolbox [1] were integrated into the implementation.

The feasibility study and experiment conducted on a prototype crime data collection application, CryHelp [73], informed the generation of more datasets for the second phase of experiment. The generation of more crime data was done using a random generator software [2] and pseudo-random algorithm based on a Gaussian distribution to populate the crime data-stream based on ground-truth provided by the users, UCT Campus Protection Service and the South African Police Service. The data are in two sets, which contain 1000 and 10 000 records respectively, since this is a reasonable bound for daily average crime report rates in South Africa [73]. In order to simulate streaming data, we used the file input stream functions in Java that enabled data to be read in real time from an external source data file into a sliding window at a random time interval bound between t_1 and t_2. The time was randomized in order to simulate a realistic crime report data stream with varying flow rates, noting that this implied some slower report arrival rates (to mimic peaceful days when there are few crime reports) and faster report arrival rates (to mimic disaster scenarios when reporting traffic is heavier).

Following is a discussion on: IL in terms of delay, IL in terms of records, gains obtained from modeling

[1] http://code.google.com/p/cse467phase3/source%20/ browse/trunk/src/Samarati.java?r=64

[2] http://www.mockaroo.com

the flow rate of the data as a Poisson process, gains obtained from incorporating three tiers of user privacy preference and using reusable anonymization clusters to reduce the number of unanonymizable/suppressed records. The experiment was done a maximum of ten times and the average of the results were noted.

6.2.1 Privacy Protection

As earlier noted, k-anonymity is susceptible to homogeneity and attribute disclosure attacks, which has the potential of leaking sensitive values. Consequently, Ashwin et al [57] came up with the concept of ℓ-diversity in order to ensure that sensitive values are diverse in each equivalence class. However, Ninghui et al [52] have proven that even ℓ-diversity is not sufficient to prevent attribute disclosure because it does not consider the distribution of the entire dataset during privacy protection. To handle this drawback, the concept of t-closeness was born. Therefore, experiments on ℓ-diversity and t-closeness are presented in this section. In addition, a comparison of the results of these three privacy schemes is shown.

Circumventing Attacks Inherent in K-Anonymity

The first experiment carried out was designed to understand the degree at which a k-anonymous dataset differs from the expected ℓ-diversity or t-closeness requirement. In other words, the degree of vulnerability of the anonymized datasets to attacks that are inherent in a k-anonymous dataset were determined. The number of records observed in each sliding window varies between five and 200 records. This implies that in an instance of time, t, when a snapshot of the data stream is taken for anonymization, the number of record varied between five and 200. This value, again, is based on the frequency of crime reported at an instance of time, t.

The attribute "crime type" was used as a sensitive attribute. Any equivalence class that has sensitive values lacking diversity is viewed as vulnerable to homogeneity attack. The value of k-anonymity was varied from the values of 2 to 4 for the first dataset and 5 to 15 for the second dataset. Our rationale for the choice of these k-values is guided by the values of k-value that are used in published experimentation results [101]. The lowest diversity level was set to 3 for all anonymization runs as a standard minimum privacy level [45]. The t value was also set to a maximum value of 0.15 in order to ensure that the distribution of sensitive values in any equivalence class is similar to the entire data stream by a maximum difference of 0.15.

From Figures 6.4 and 6.5, it was observed that a lower k-value is more susceptible to attacks such as homogenity and attribute disclosure, which are inherent in k-anonymity and anonymized datasets that are not ℓ-diversed and t-closed. It was also observed that as the privacy value of k-anonymity is increased, the degree of susceptibility of such attacks also begins to reduce.

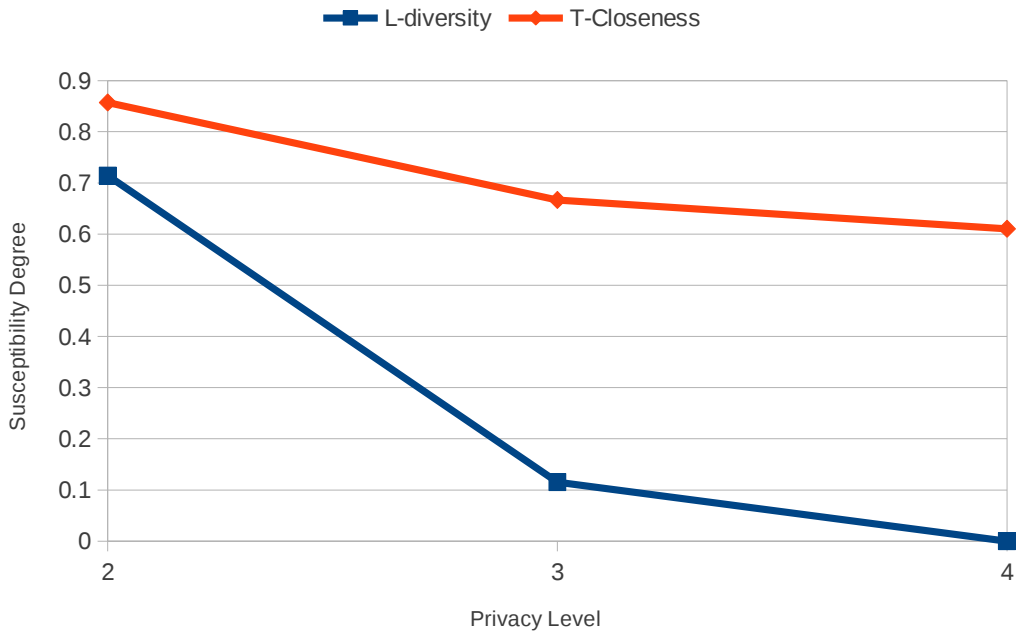


Figure 6.4: Effect of increase in k-anonymity Privacy Level (k) on Homogeneity and Attribute Disclosure Attack for Dataset 1

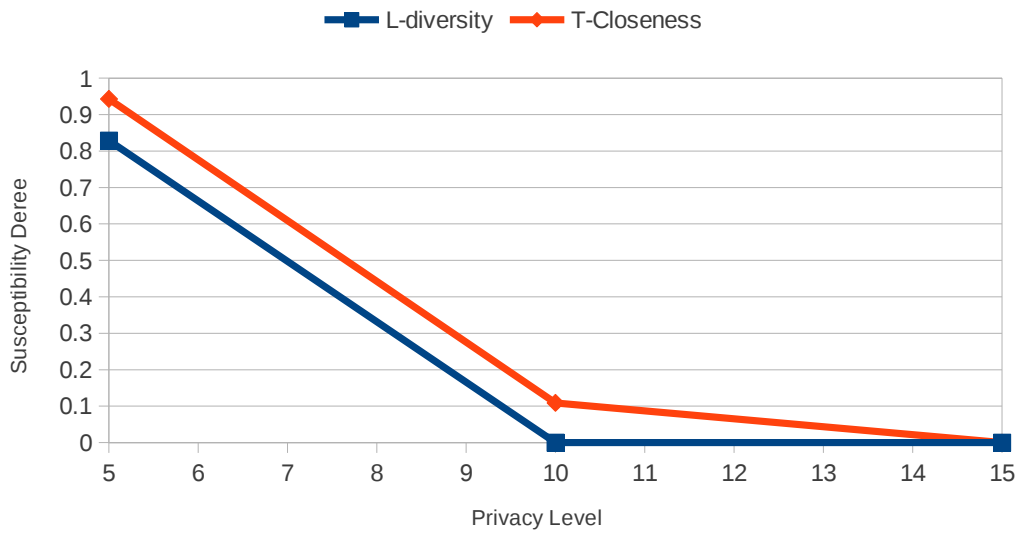


Figure 6.5: Effect of increase in k-anonymity Privacy Level (k) on Homogeneity and Attribute Disclosure Attack for Dataset 2

Data Utility and Information Loss

IL simply means the deviation of anonymized data from the initial form. In Chapter 2, different metrics that can be used to measure IL were discussed. For this research, the generalized loss metric [38] was adopted because it is a benchmark in many data stream anonymization schemes [10, 29, 101]. The total IL of a data stream was calculated by averaging the IL of all records in it.

According to Figures 6.6 and 6.7 k-anonymity generally has the lowest IL of the three privacy schemes. However, it was observed that as the privacy value of k is increased and the privacy value of ℓ and t is maintained, the IL of the three schemes tends to be uniform/balanced because a higher k-value increases the number of possible records in an equivalence class, which leaves room for more diversification and uniformity. Consequently, the higher the k-value, the higher the likelihood of a well-diversified and evenly distributed equivalence class. Overall, it can be concluded that higher privacy and utility are achieved if the three privacy measures are combined.

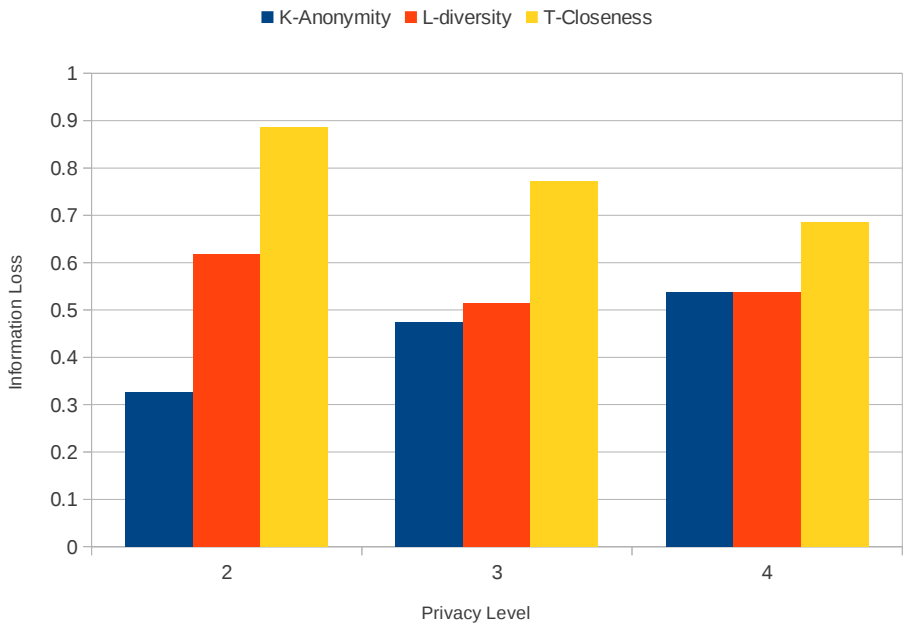


Figure 6.6: Effect of different Privacy Schemes (that is, k-anonymity, ℓ-diversity and t-closeness) on Information Loss, k = 2 - 4, l = 3, t = 0.15 for Dataset 1

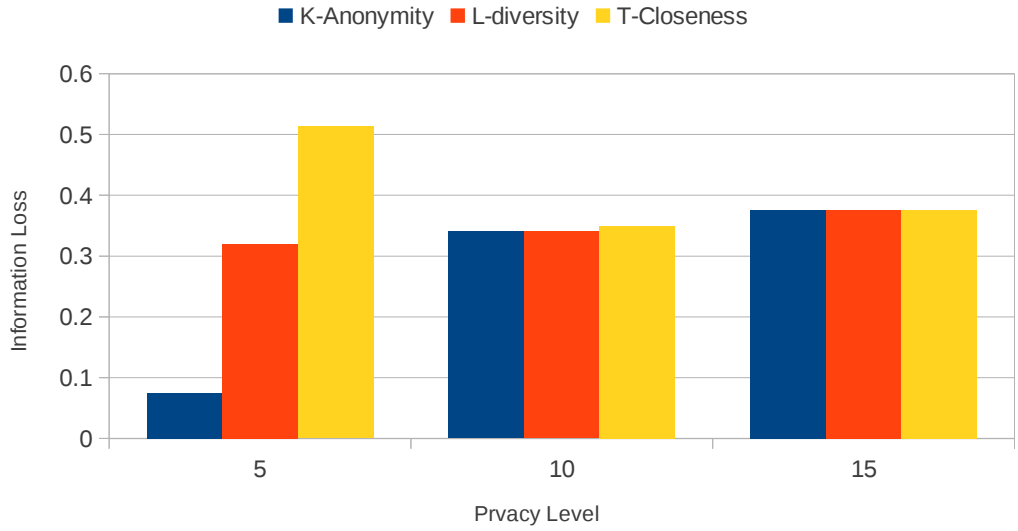


Figure 6.7: Effect of different Privacy Schemes (that is, k-anonymity, ℓ-diversity and t-closeness) on Information Loss, k = 2 - 4, l = 3, t = 0.15 for Dataset 1

Privacy versus Execution Time

Execution time is used as a good measure of performance. Of interest is seeing how execution time could vary at implementation level based on different privacy requirements [8], hence the rationale for this experiment. The execution time of the three privacy measures, namely; k-anonymity, ℓ-diversity and t-closeness, were compared.

Pre-experiment sampling revealed that there were instances where ℓ-diversity or t-closeness resulted in poor performance or high IL just because one or a few equivalence class(es) were not well diversified or distributed. Therefore, two thresholds were introduced, α and β, for ℓ-diversity and t-closeness, where $\alpha \leq 1$ and $\beta \leq 1$. The purpose of these thresholds is to determine if an initial result obtained from ℓ-diversity and t-closeness satisfies the privacy constraint to at least a degree of $1 - \alpha$ and $1 - \beta$. If these thresholds are not met, then advanced ℓ-diversity and advanced t-closeness take place, which typically works by combining equivalence classes or moving up the hierarchy tree to find a solution, usually at higher computational cost and of poorer utility. In summary, base ℓ-diversity can be defined as a post-activity of the k-anonymity process to check for satisfaction of ℓ-diversity to at least α degree, while advanced ℓ-diversity takes place if base ℓ-diversity fails, by either combining equivalence classes or searching the solution space again for a solution that satisfies both k-anonymity and ℓ-diversity. The same concept applies to base t-closeness and advanced t-closeness.

Figure 6.8 shows the execution time for dataset 1 with $2 \leq k \leq 4$, $\ell = 3$, t = 0.15, $\alpha = 0.1$ and $\beta = 0.1$, while Figure 6.9 shows the execution time for dataset 2 with $5 \leq k \leq 15$, $\ell = 5$, t = 0.10, $\alpha = 0.1$ and $\beta = 0.1$. As shown in the figures, ℓ-diversity has the shortest execution time compared to the other measures. This is mainly because ℓ-diversity runs just to check if the sensitive attribute is diversified according to some parameters.

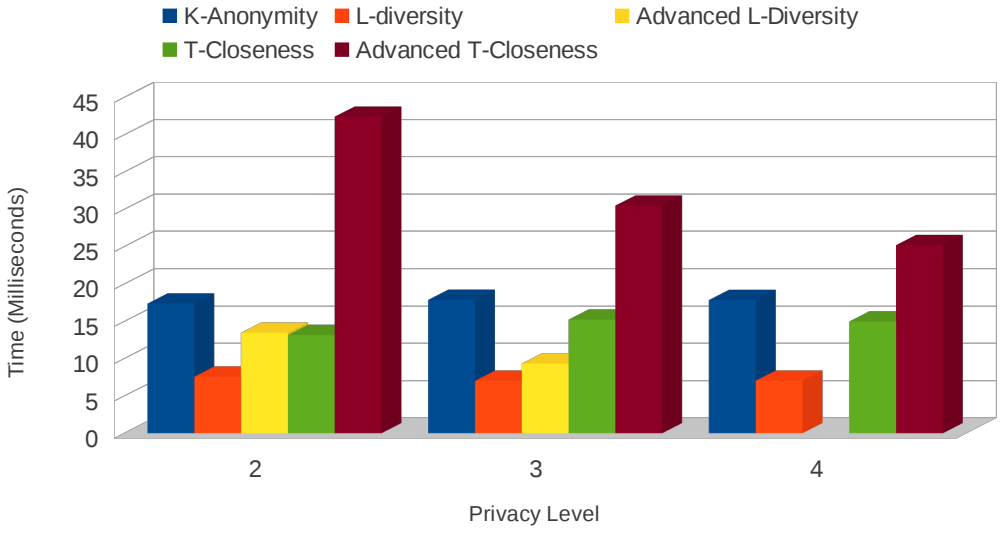


Figure 6.8: Execution Time Versus Privacy Scheme for Dataset 1; k = 2-4, $\ell = 3$, t =0.15

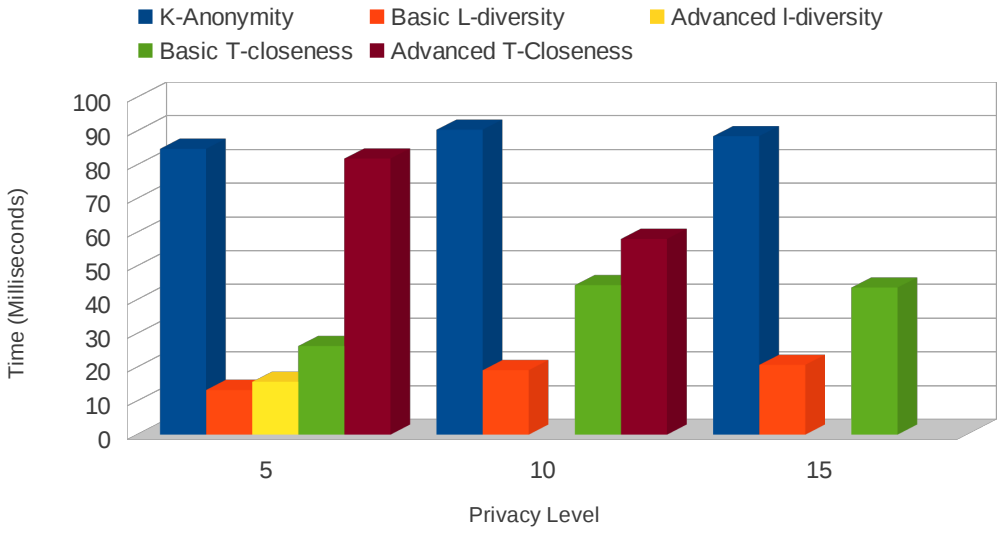


Figure 6.9: Execution Time Versus Privacy Scheme for Dataset 2, k = 5-15, $\ell = 5$, t =0.10

6.2.2 Gains Obtained from Modeling the Flow Rate of the Data as a Poisson Process

Having established that the use of the three privacy schemes; k-anonymity, ℓ-diversity and t-closeness, guarantees a high degree of anonymization, we then proceeded to carrying out experiment on how the use of Poisson probability distribution can help to reduce IL. Poisson probability distribution was employed to study the rate at which data flows in the stream and subsequently to determine the appropriate size or time a sliding window, sw_i, should exist for taking record of arrival rates and unanonymizable records into consideration.

Therefore, this section discusses the gains obtained using Poisson probability distribution to predict the time a sliding window should exist, while ensuring that records do not expire and that the number of unanonymizable (or suppressed) records is minimal. The gains obtained are explained in the following sub-sections:

Recovered Unanonymizable Tuples

During anonymization there is usually a trade-off between the rate of IL, suppression and generalization. Usually if an equivalence class is unable to satisfy the privacy requirement, such a class is merged with another class or all its records are suppressed. A higher suppression rate implies that vital information is likely to be concealed from the recipient of the anonymized table, while merging of classes implies an increase in IL, which has the drawback of offering lower data utility. In order to curb this, Poisson probability distribution predicts the chances of such unanonymizable (suppressed) records undergoing anonymization in the next sliding window in a manner that preserves privacy and maximizes data utility with the goal of minimizing delay or expiration of records.

Figures 6.10 and 6.11 show the rate at which unanonymizable records were anonymized again, going by the predictions of Poisson probability distribution. It is evident from the figure that many unanonymizable records were recovered and allowed to go for anonymization again. It was also observed that the probability threshold influenced the number of unanonymizable records recovered. This leads to the conclusion that the higher the probability threshold, the lower the probability of unanonymizable records being given a chance for anonymization re-consideration in subsequent sliding window(s). The implication of this is that more records are likely not to be given the chance of another round of anonymization if higher threshold values are used. Another observation is that if a higher threshold value is used, then

there are fewer changes or movements in records between sliding windows.

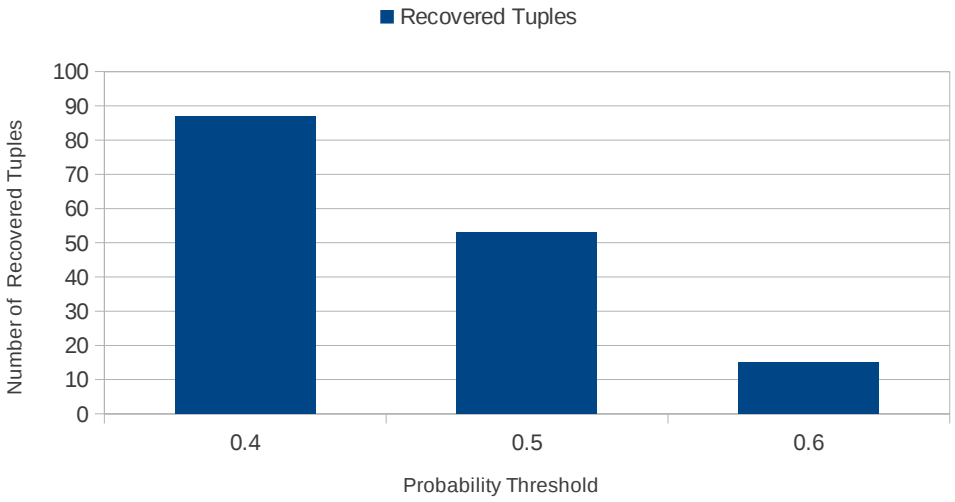


Figure 6.10: Poisson Probability Threshold Versus Recovered Tuples for Dataset 1

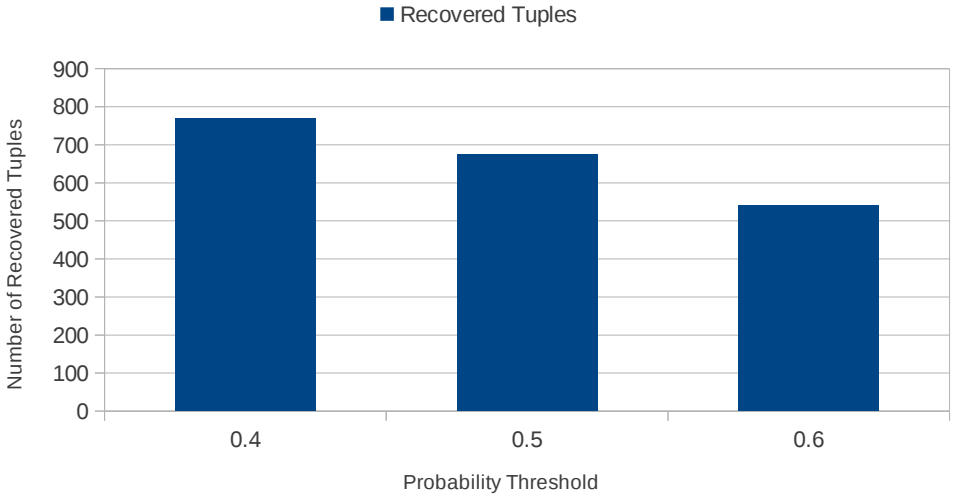


Figure 6.11: Poisson Probability Threshold Versus Recovered Tuples for Dataset 2

Privacy Value/Level versus Recovered Unanonymizable Tuples

"Privacy level" simply means the degree of anonymity offered, while unanonymizable tuples are those tuples that belong to an equivalence class whose size is less than k. For the purpose of sliding windows that start with a small number of tuples, the minimum privacy level threshold was set as k=2 and the maximum at k = 15; the ℓ-value was varied between values 3 and 5 and finally the t-value was alternated between values 0.1 and 0.15 for the two datasets.

As illustrated in Figures 6.12 and 6.13, it was observed that as the privacy value/level increases, the possible number of unanonymizable records that can be recovered using Poisson probability prediction is reduced. The main reason for this is that as the privacy level or degree increases, it is expected that the rate or possibility of achieving anonymization will become increasingly challenging. This definitely also influences the expectation of higher chances of anonymization rate for unanonymizable records. To understand the reason for the decline in the rate of recovered unanonymizable records better, let us assume the k privacy level is set to three and an equivalence class, EC_i, has two records; this implies that at least one more record is needed to make EC_i satisfy k-anonymity. In essence, using Poisson probability, the model attempts to predict the chance of at least one record in EC_i arrive within time, t, in the next sliding window. If k is set to four, this will mean the chance of at least two records arriving in the next sliding window. An implication of this is that the chances of having at least two records is more difficult or demanding compared to the chances of just one record. Thus, this explains why the model has a drop in recovered unanonymizable records as privacy level increases. Therefore, the conclusion is that the rate at which unanonymizable records in a current sliding window can be anonymized in a subsequent window is mainly dependent on the privacy value.

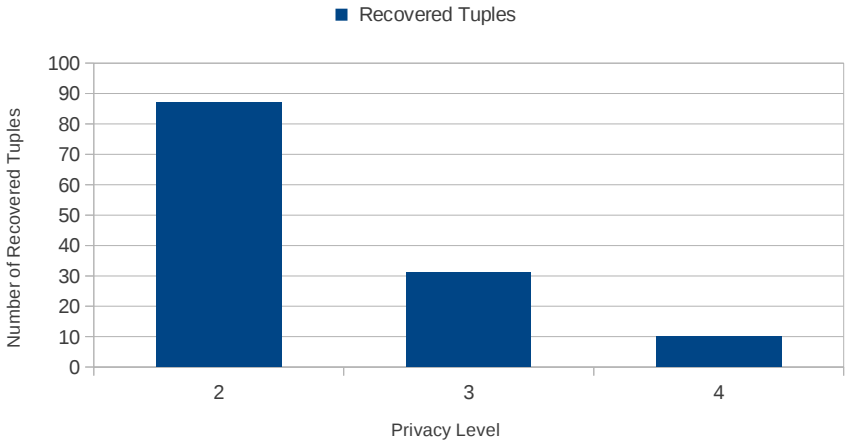


Figure 6.12: Relationship Between Privacy Level and Recovered Tuples for Dataset 1

Effect of Time-Based Sliding Window on Information Loss (Records):

In order to measure the effect of the time-based sliding window on IL, the k-value is set to values between 2 and 4, δ, i.e. the Poisson probability threshold, to 0.4, and time-based sliding window to values between 2000 ms and 5000 ms. The choice of $t_l = 2000$ ms and $t_u = 5000$ ms, is guided by values of delay that are used in published experimentation results [101]. The choice of $\delta = 0.4$ is based

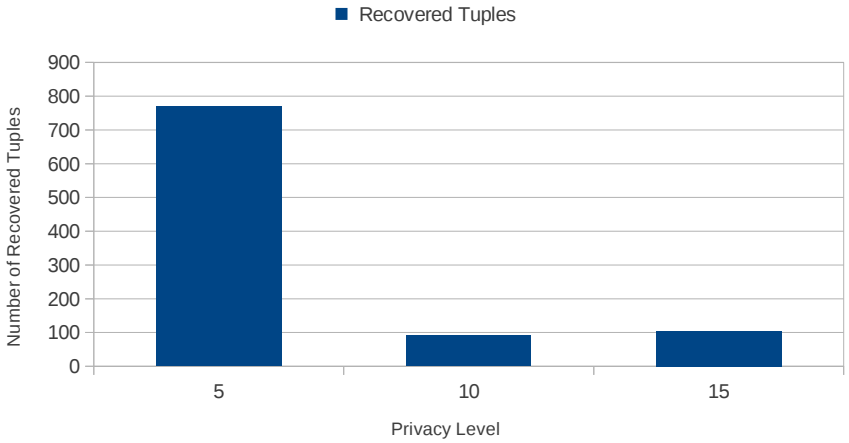


Figure 6.13: Relationship Between Privacy Level and Recovered Tuples for Dataset 2

on the various experiments conducted. Thus, varying δ from 0.4 to 0.6 achieved the best output at 0.4.

To calculate IL with respect to the number of records i.e. deviation of anonymized data from its initial form, we used the formula in equation 4.5, as it is in [38]. We adopted this metric because it is a benchmark in many data stream anonymization schemes [10, 29, 101].

$$\text{InfoLoss} = \frac{\text{M}_P - 1}{\text{M - 1}} ...(5)$$

M_p is number of leaf nodes in the subtree at node P and M is the total number of leaf nodes in the generalization tree. The IL of a sliding window, $SW_i = \{\text{R}_1, \text{R}_2, \text{R}_3, ..., \text{R}_n\}$ is calculated as follows:

$$\frac{1}{\text{n}} \sum_{i=1}^{n} \text{InfoLoss}(R_i) ...(6)$$

The total IL of a data stream is simply calculated by averaging the IL of all sliding windows in it.

Figure 6.14 shows the effect of applying the time-based sliding window buffering mechanism and Poisson probability distribution model to IL. Here, it is observed that for smaller sliding window sizes IL is lower in comparison to larger window sizes. One of the reasons for this is that most often, records in lower sliding windows are usually fewer and as a result have tendency to be suppressed due to insufficient records to form appropriate clusters necessary for anonymization. Since the solution mainly focuses on reducing the rate of suppressed (or unanonymizable records), the Poisson distribution and reusable clusters becomes more active in the lower sliding windows. This helps to reduce IL in lower sliding

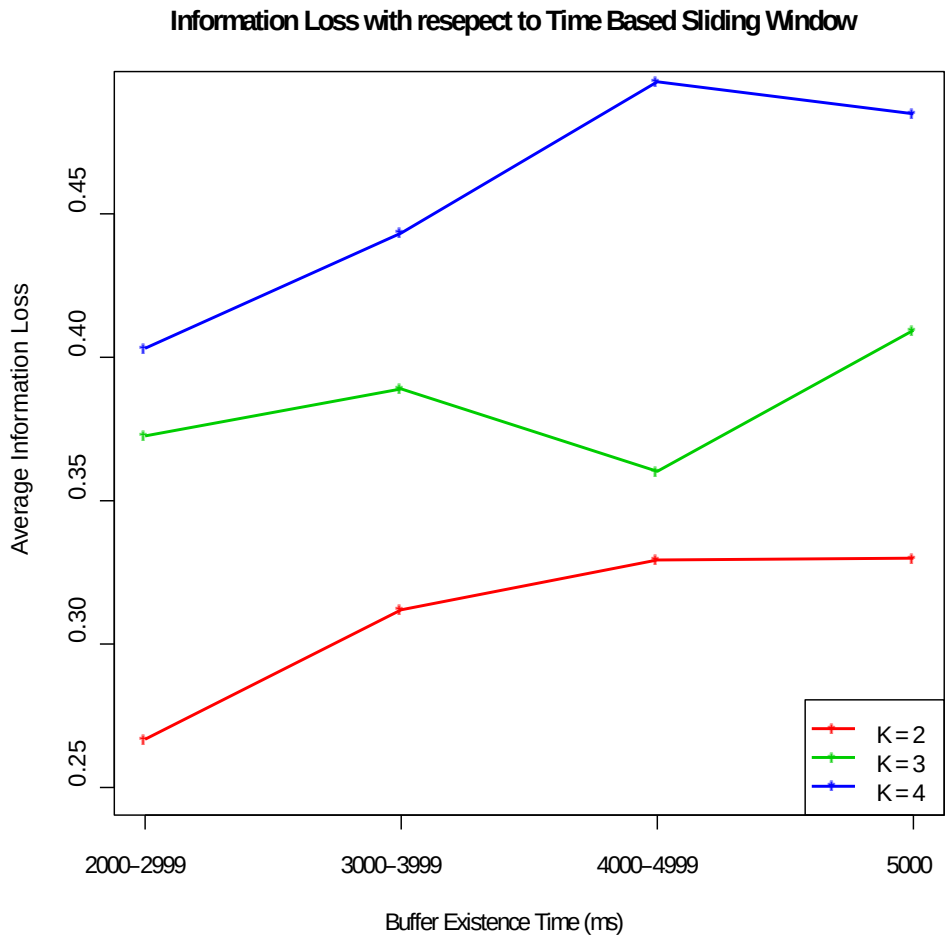


Figure 6.14: Effect of Sliding Window Size and Privacy Level Variation (Expressed in terms of k-value) on Information Loss

window sizes.

It is also observed that as the anonymity degree increases, privacy is enhanced and the anonymization quality or output drops. It therefore implies that an increase in privacy level, k, also leads to an increase in IL.

Record Suppression

One of the goals of a good anonymization scheme is to ensure that IL is minimal. Records suppression happens in cases of outliers and this usually leads to high IL. The combination of the reusable cluster and the Poisson distribution helped to minimize the total number of suppressed records and as a result

reduced IL. However, our approach was unable to effectively recover some of the suppressed records because their deadlines had already been exceeded or the Poisson probability prediction for recovering those records was low and a suitable reusable cluster could not be constructed before the record expired.

As shown in Figure 6.15, it is observed that as k-value increases, the reusable cluster is more active and as a result there is an increase in records that would have been suppressed but are allowed to be anonymized in a manner that yields a lower information loss.

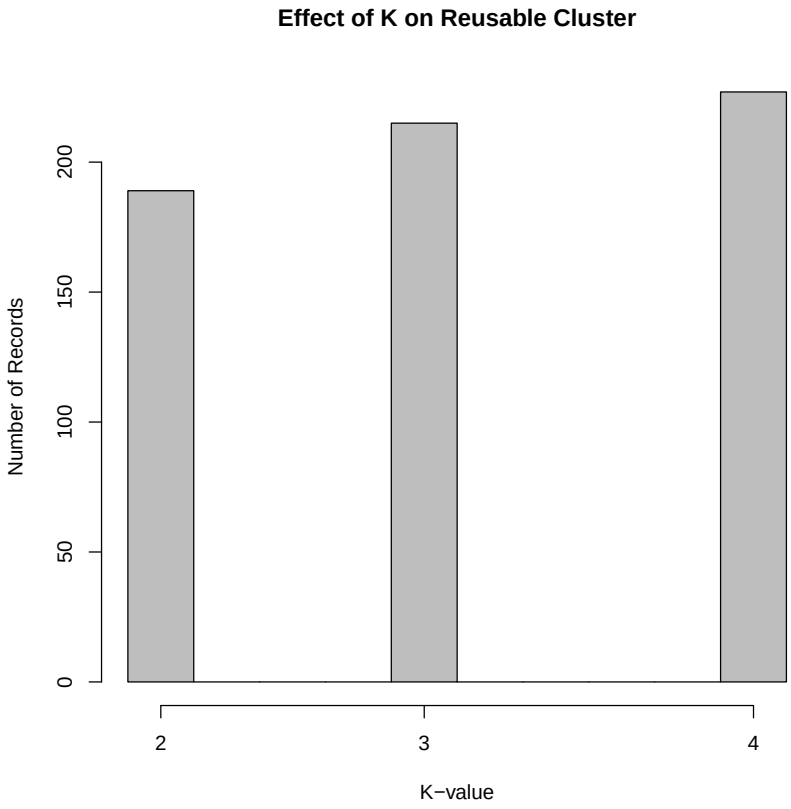


Figure 6.15: Impact of the Reusable Cluster on Recovering Records

6.2.3 Benchmarking: Poisson Solution Comparison with Non-Poisson Solution

As a baseline case for evaluating the proposed adaptive buffering scheme, the proactive-FAANST and passive-FAANST, were implemented. These algorithms are a good comparison benchmark because they are the current state-of-the-art streaming data anonymization and reduce IL with minimum delay and expired tuples [101]. The proactive-FAANST decides if an unanonymizable record will expire if

included in the next sliding window, while passive-FAANST searches for unanonymizable records that have expired. A major drawback of these two variants is that there is no way of deciding whether or not such unanonymizable records would be anonymizable during the next sliding window. This is necessary to avoid repeatedly cycling a tuple that has a low chance of anonymization in subsequent sliding window(s). Moreover, these algorithms do not consider the fact that the flow or speed of a data stream could change. These weaknesses of proactive-FAANST and passive-FAANST are what we attempt to address by using Poisson probability distribution to predict if such tuples would be anonymizable in subsequent sliding window(s) by taking into consideration the arrival rate of records, success rate of anonymization per sliding window, time a tuple can exist and rate of suppressed records.

Expired Tuples: Information Loss in Delay

A tuple expires when it remains in the system for longer than a pre-specified threshold called delay [101, 59]. In order to decide whether a tuple has exceeded its time-delay constraint, additional attributes such as arrival time, expected waiting time and entry time were included. As a heuristic, the choice of delay values, $t_l = 2000$ ms and $t_u = 5000$ ms, is guided by values of delay that are used in published experimentation results [101].

The sliding window size for the Poisson solution varies between t_l and t_u. The window size for a passive and proactive solution in the experiment was chosen to be eight records. The choice of this value was based on the number of records that arrive in the slow data stream within 5000 ms. Within 5000 ms, few as six to eight records and as many as 20 records were observed. Therefore, eight records were chosen as the window size to minimize expired tuples.

In general, the approach shows that there are fewer expired tuples when compared to passive-FAANST and proactive-FAANST solutions. This is because before our Poisson prediction transfers suppressed records to another sliding window, it checks for the possibility of their anonymization. In other solutions, there is no mechanism in place to check the likelihood of the anonymizability of a suppressed record before allowing it to go to the next sliding window/round. As a result, such tuples are sent to the next sliding window and have high a tendency to expire eventually.

Our solution also shows that the lower a k-value, the higher the number of expired tuples. This is because the outcome of Poisson prediction is lower for higher k-values. As a result, there are fewer changes of sliding windows as the k-value increases and this means there is a lower possibility of expired

tuples.

One of the main goals of this solution is to reduce IL in delay (i.e. to lower the number of expired tuples). Figure 6.16 depicts that the solution is successful in achieving its main goal, and the IL (delay) in the solution is lower than in passive and proactive solutions. In order to determine the total number of records that expired, a simple count function was used to retrieve all records that had remain in the buffer longer than the upper limit threshold, t_u. To determine the average expired records, the expired records in all the experiments were summed up and divided by the total number of experiments.

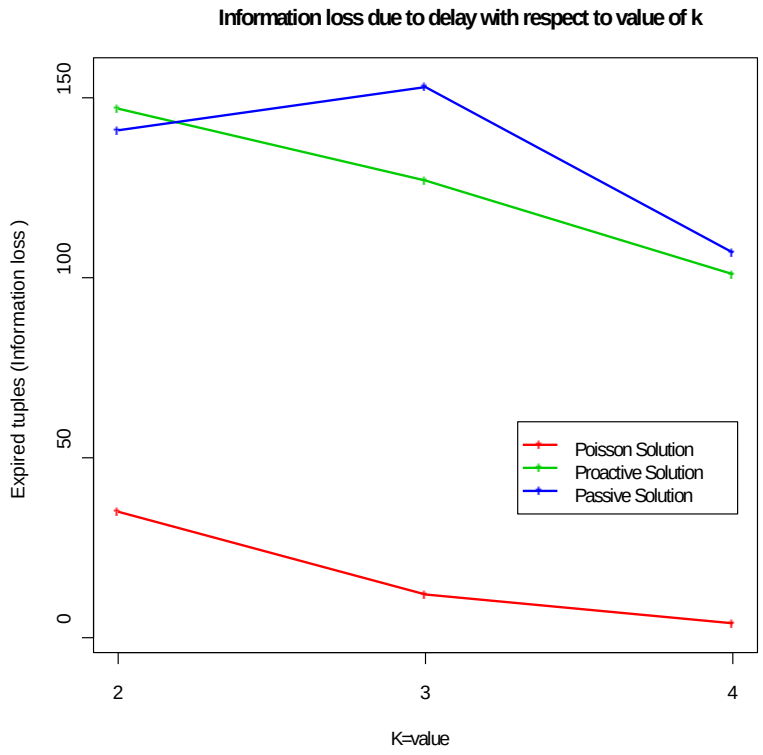


Figure 6.16: Privacy Level Versus Expired Tuples for Poisson Solution, Passive-FAANST and Proactive-FAANST

Data Utility: Information Loss in Record

An important factor that is considered in anonymization is the degree of usability of anonymized data for data analysis or data mining tasks [55]. Therefore, comparison of the degree of IL of our solution with that of Passive-FAANST and proactive-FAANST was made. The result, as illustrated in Figure 6.17, shows that at the minimal level of privacy enforcement, the information loss of the solution is on par with the other two schemes, while at the maximal level our solution has better data utility.

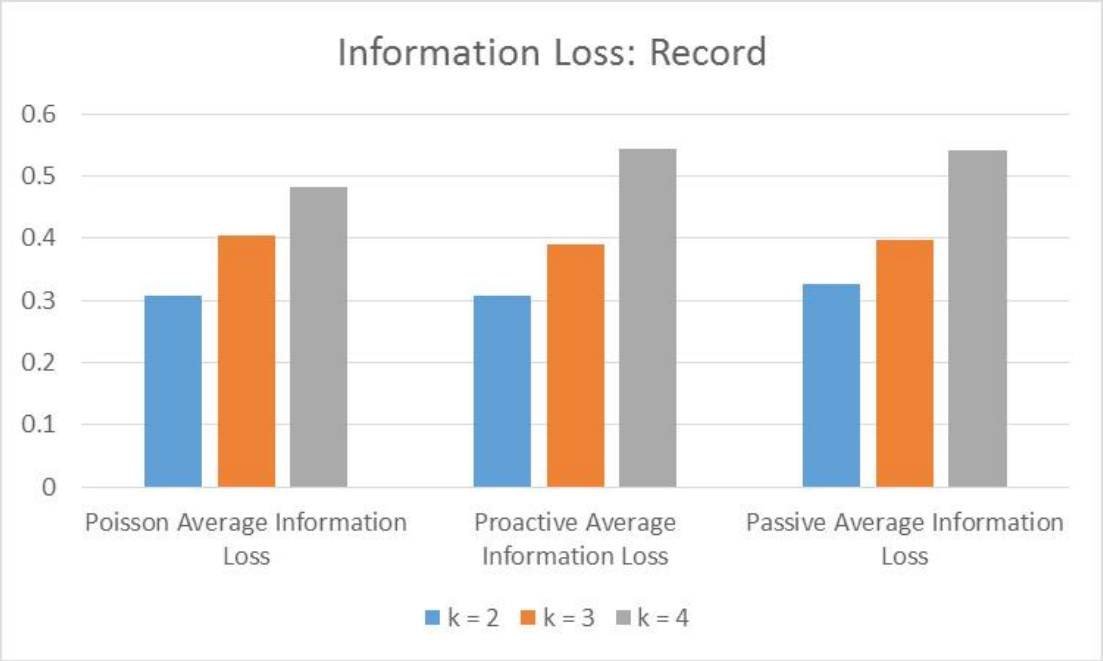


Figure 6.17: Privacy Level Versus Information Loss for Poisson Solution, Passive-FAANST and Proactive-FAANST

6.3 Experiment on User-Defined Privacy Preferences

The three-tier user-defined privacy preference was integrated into k-anonymity by starting with a general k-value and progressing to a more specific or personalized k-value. As a heuristic, the choice of a general k value is guided by values used in published experimentation results [101].

As stated earlier, k-anonymity uses a generic approach to enforce privacy preservation for all users without catering for their concrete needs. The outcome of this is that insufficient protection might be provided to a subset of people, while excessive privacy control is provided to another subset. Therefore, this

experiment is intended to ensure that there is balanced protection by taking users' privacy preferences into consideration.

6.3.1 Reduction of Excessive Privacy Control:

The results of experiment, as reflected in Figure 6.18, show that integration of our approach into k-anonymity, in comparison to other approaches, ensures that excessive privacy control is reduced, while at the same time guiding against insufficient protection. The y-axis, which is labeled number of records, refers to those records that were excessively protected when compared to user's personal preference. This three-tier personalised approach has a 16.15% rate of excessive privacy control, while the Personalised model suggested by Gedik [25] and the non-personalised model have a 63.08% and 23.08% rate of excessive privacy control respectively.

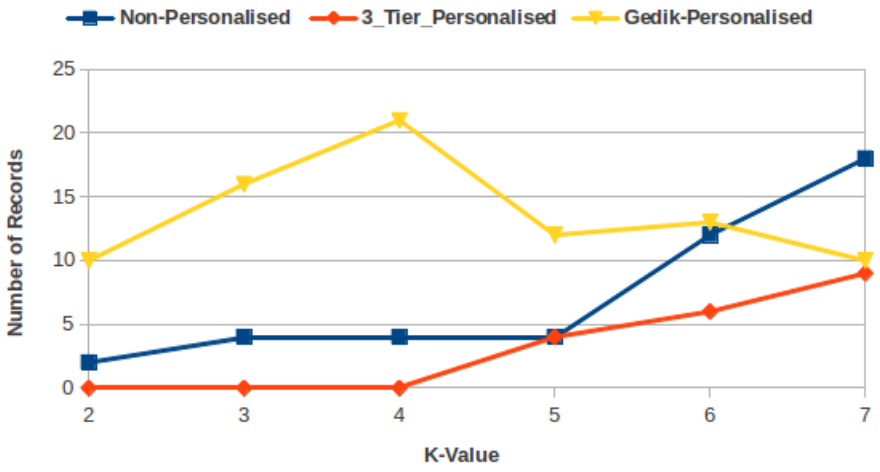


Figure 6.18: Effect of Personalised and Non-Personalised Privacy on Excessive Privacy Control

The reason that this approach performed better than that of Gedik and the non-personalized privacy model is that a general k-value was first used and personalization is only required when there is excessive privacy control in comparison to users' preference. The result of the three-tier personalized result also shows that the higher the k-value, the higher the rate of excessive privacy control. This is because as the k-value increases anonymization and the privacy quality increase too. Hence, more records have the chance of being suppressed, which in turn leads to excessive privacy control.

6.3.2 Record Suppression:

One of the goals of a good anonymization scheme is to ensure that IL is minimal. Record suppression usually leads to high IL. The use of a personalized privacy scheme minimizes the total number of suppressed records and as a result reduces IL while the use of non-personalized privacy scheme leads to a high number of suppressed records.

Figure 6.19 shows the effect of personalized and non-personalized privacy on suppressed records. The y-axis, which is labeled records, refers to the total number of records that were suppressed. The result shows that our three-tier personalized privacy model has a lower rate of suppressed records, 26%, when compared to the Gedik personalized privacy model that has 77.7% and the non-personalized privacy model that has 51%. This is because the three-tier personalized privacy model considers suppressed records and attempts to reduce the number using users' privacy preferences. In particular, this reduction is carried out by the level 2 protection scheme described in section 5.6 of Chapter 5 .

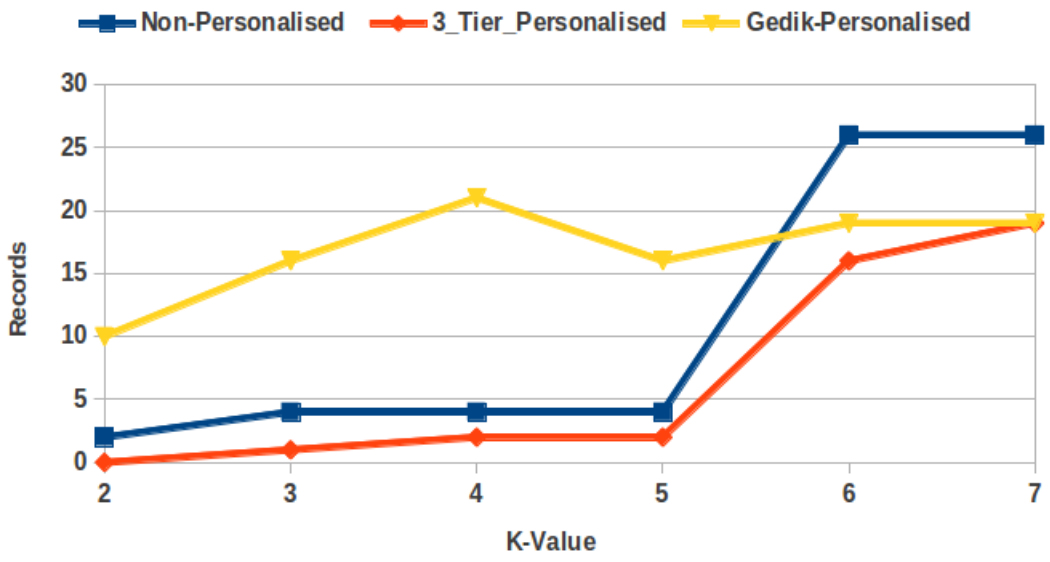


Figure 6.19: Impact of the Personalized and Non-Personalized Privacy Scheme on Minimizing Number of Suppressed Records

6.3.3 Computation Cost

Figure 6.20 shows the effect of the three-tier, the Gedik personalized and non-personalized privacy on computational cost. The figure shows that our approach has the highest computation cost, while the

Gedik personalized privacy model has the lowest computation cost.

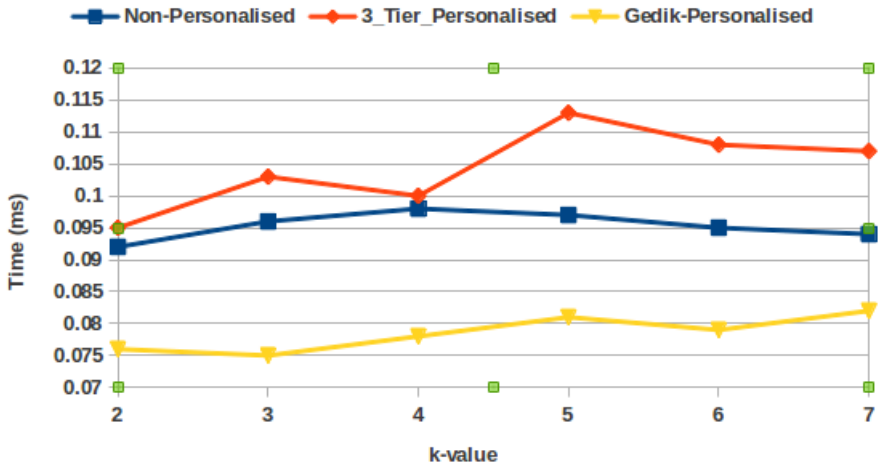


Figure 6.20: Impact of the Personalized and Non-Personalized Privacy Scheme on Computation Cost

Our approach has the highest computation cost because it searches for over-protected and suppressed records in order to ultimately offer a lower IL, while the Gedik personalized and the non-personalized approach model does not search for suppressed and over-protected records. Our result also shows that the higher the k-value, the higher the computation cost.

6.4 Chapter Summary

This chapter presented experimental results using both real and quasi-real data. The quasi-real data were necessary to test the scalability of our solution. Results from the first set of experiments revealed the potential usefulness of CryHelp as a useful tool for a secure crime report in a resource-constrained environment. Afterwards, the results of the anonymization experiments carried out on the crime data stream were discussed. The experiment was carried out in a progressive manner by starting with the fundamental privacy model and subsequently refining its performance to mitigate any form of vulnerability. The final results obtained as a result of the refinement revealed that the combination of the three privacy schemes (k-anonymity, ℓ-diversity and t-closeness) offers a solution that is privacy-preserving. Then there was a discussion on the benefits that were gained as a result of the integration of the ABRS model. Furthermore, a comparison of the model (ABRS) with common (baseline) data stream delay-reduction techniques, proactive-FAANST and passive-FAANST, was presented. Experimental results reveal that these techniques are generally not able to reduce IL as ABRS can. These results demonstrate the re-

liability of ABRS in improving performance during data stream anonymization with specific focus on the reduction of IL and record expiration. Lastly, wan experiment was carried out to measure the effect of a user-defined privacy preference on the reduction of excessive privacy control, record suppression the three-tier user privacy preference outperforms the non-personalized privacy model and some existing personalized-privacy schemes.

Chapter 7

Conclusion and Future Research

7.1 Introduction

Numerous studies have been carried out in the field of data privacy and anonymization. Of interest are studies on how to protect and anonymize streaming data. One challenge that anonymization of streaming data faces is that of high levels of IL and delay during anonymization. In addition, the existing schemes and techniques that have been used to combat this challenge do not adequately address the problem. Hence, there is a need for more research in an effort to combat this challenge.

As an intervention strategy, this research developed a crime-reporting application system (CryHelp App) which enables crime to be reported in a secure and covert manner. Using the data from CryHelp App, this research attempts to address the challenge of IL while adequately protecting data. Furthermore, two models were built to reduce IL and provide adequate protection during anonymization. The first model uses Poisson probability distribution and time-based sliding windows to adaptively resize a buffer based on the arrival of crime reports, and thus reduces loss of information considerably. The second model is built on the concept of a three-tier user-defined privacy scheme, which anonymizes records based on users' preference.

7.2 Summary

This thesis began with an explanation of the problem scenario that emerges in resource-constrained environment where the lack of data analyst expertise in law enforcement agencies results in the need for a third-party data analyst provider to aid in fast (crime) report analysis for knowledge support. In addition, there was a highlight on the fact that the growing need to make the processed information available to field officers requires a mechanism for capturing crime reports in real time and transferring these reports to the third-party service provider. While solutions in the literature based on cryptography have been shown to be successful in protecting data in outsourced scenarios from unauthorized access, including that of “honest-but-curious" service providers, it was noted that querying encrypted streaming data is a time-consuming process and that the anonymization technique is a more practical approach to data privacy preservation in this case. However, the generic paradigm approach to privacy enforcement in the anonymity model needs to be refined in order to cater for individual needs. As a result, this research emphasized the need to integrate users' privacy preference into anonymization while attempting to reduce delay caused by buffering mechanisms.

To address the aforementioned challenges, this study is based on three main research questions, namely:

- How can crime be reported in a secure and covert manner?
- How can an anonymization scheme such as k-anonymity and its variants support data stream anonymization in a manner that reduces IL and expired records?
- How can an anonymization scheme or model capture users' privacy preference?

This chapter concludes the thesis by presenting a summary of the research work, suggesting recommendations and presenting opportunities for future research. A synthesis of how the research questions were addressed follows.

7.3 Synthesis of Empirical Findings

7.3.1 How can Crime be Reported in a Secure and Covert Manner?

In order to address the first research question, a survey was conducted among students at the University of Cape Town who identified that a mobile phone provides a platform by which crime could be reported in a usable and secured manner. Afterwards, users were able to come up with prototypes of how the intended crime-reporting application should be designed. These prototypes went through different iterations of implementation in order to develop the intended CryHelp App. The final output of the iterations was deployed to users to test it for usability. The deployment and evaluation results of the CryHelp App show that overall the system was well received, with a score of 77.06%. Further results show that the interface quality (78.33%), though marginally, is the most appreciated aspect of the system, as the design process was centered on the users. These results are promising for the feasibility of a mobile solution for crime reporting in resource-constrained environment.

7.3.2 How can an Anonymization Scheme such as k-anonymity and its Variants Support Data Stream Anonymization in a Manner that Reduces Information Loss and Expired Records?

This research addressed the way in which streaming crime data can be anonymized using k-anonymity, ℓ-diversity and t-closeness such that expired records (delay) and IL are minimized. In order to do this, the streaming data was modeled as a flow that follows Poisson probability distribution. The buffer was also modeled as a time-based tumbling sliding window in order to ensure that no record exceeds its deadline by allotting a time limit to each sliding window. The Poisson probability distribution determines the best time to allot based on the arrival of data in the stream. The experimental results show that in addition to ensuring the privacy of the data, the proposed scheme outperforms others with an IL rate of 1.95% in comparison to 12.7% on varying the privacy level of crime report data records.

7.3.3 How can the Anonymization Process Capture Users' Privacy Preference?

Finally users' privacy preference was incorporated into the anonymization scheme. This preference is classified as low, neutral or high. In order to automatically guide users' choice of privacy preference,

multinomial regression and association rule models were used to predict a user's privacy preference. The basis for this is that a reporter is likely to be under duress during crime reporting and it might be best to guide the model to choose an appropriate privacy preference automatically. The use of three-tiered user-defined privacy when integrated into the association rule and multinomial regression reduces over-protection compared to other existing models. Over-protection occurs when a privacy level greater than a user's need is used for anonymization. Our three-tiered personalized approach has a 16.15% rate of over-protection, while the personalized approach proposed of Gedik [25] and non-personalized approach of Sweeney [85] have a 63.08% and 23.08% rate of over-protection respectively. Therefore incorporating the three-tier privacy level approach into the anonymization model is an improvement on existing models.

7.4 Limitations of Research

The CryHelp App, being a prototype, could be improved upon to fit a wider range of data. While the application is effective and efficient in capturing crime details, it has the limitation of not being able to capture several images, audio and video clips of crime reports. The anonymization of this type of data was consequently not considered.

This research uses CryHelp as a proof of concept solution that focuses on anonymizing streaming crime data using viable means. The research did not compare the CryHelp App to other existing crime-reporting applications, although the quality of results and the level of findings obtained in the analysis show that the proposed model is promising for streaming data anonymization.

7.5 Potential Extensions and Future Research

7.5.1 Other Forms of Data

CryHelp only considered text data for anonymization. However, in today's information age where data could come in different formats and in high volume, it is necessary to consider anonymizing image and video data.

7.5.2 Diverse Dataset

The system developed in this research is a proof-of-concept solution that focuses on the anonymization of crime streaming data. There is great potential for extending the current research to handle different datasets. Future research could consider evaluating the models on de facto anonymization benchmarks such as the adult's census dataset from the UC Irvine machine learning repository.

7.5.3 Longitudinal Deployment and Evaluation

The study is based on an application in one school location (UCT), thereby excluding a wider range of testing in different schools of the cities in developing countries in general. A possible extension will be to consider more universities in developing nations. Such an extension would also provide more datasets for the evaluation of the anonymization model and could eliminate the need for the generation of quasi-real data.

Appendix A

Data Description and Survey Overview

A.1 Overview of CryHelp Data

```
<?xml version="1.0" encoding="UTF-8"?>
- <UserReport>
  - <User>
      <Name> Anonymous</Name>
      <Surname> </Surname>
      <ID_Number> </ID_Number>
      <Student_Number> </Student_Number>
      <Address> </Address>
      <Cell_Number> </Cell_Number>
      <Home_Number> </Home_Number>
      <Level_of_Privacy> 1</Level_of_Privacy>
    </User>
  - <CrimeData>
      <Time_of_occurrence> </Time_of_occurrence>
      <Date_of_occurrence> </Date_of_occurrence>
      <Place_of_Occurrence> </Place_of_Occurrence>
    </CrimeData>
  - <CrimeDetail>
      <Brief_Detail_of_the_Offence> </Brief_Detail_of_the_Offence>
      <TAGS> </TAGS>
    </CrimeDetail>
  - <SuspectData>
      <Student_Number> </Student_Number>
      <Name> </Name>
      <Telephone_Number> </Telephone_Number>
      <Address_Bus_Res_> </Address_Bus_Res_>
    </SuspectData>
  - <SuspectDetail>
      <Male_Female> </Male_Female>
      <Height> </Height>
      <Weight> </Weight>
      <Age> </Age>
      <Build> </Build>
      <Hair> </Hair>
      <Face> </Face>
      <Eyes> </Eyes>
      <Complexion> </Complexion>
      <Black_Coloured_White> </Black_Coloured_White>
      <Wearing> </Wearing>
      <Exhibits_items_on_scene_> </Exhibits_items_on_scene_>
    </SuspectDetail>
</UserReport>
```

Figure A.1: Overview of Details Collected Using the CryHelp App

A.2 CryHelp User Interface Questionnaire

Gender:

Age:

Pre-test Questions

Are you a student?

Have you ever reported a crime before?

Do you possess a cellular phone?

How often do you use your phone?

How familiar are you with the android OS?

Scenario Questions 1

For each of the items below, please circle the response that best describes your experience with the application for this scenario.

1.Sending a full crime report

Time to Complete Task

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

Ease of Performing Tasks

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

2. Taking an image of a scene

Time to Complete Task

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

Ease of Performing Tasks

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

3. Tagging the image

Time to Complete Task

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

Ease of Performing Tasks

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

4. Inputting gesture

Time to Complete Task

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

Ease of Performing Tasks

ACCEPTABLE AS IS 1 2 3 4 5 NEEDS A LOT OF IMPROVEMENT

Scenario Questions 2

For each of the items below, please circle the response that best describes your experience with the app for this scenario.

1. Overall, I am satisfied with how easy it is to use this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

2. It is simple to use this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

3. I can effectively complete a crime report using this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

4. I am able to report a crime quickly using this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

5. I am able to submit a crime report efficiently using this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

6. I feel comfortable using this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

7. It was easy to learn to use this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

8. I believe I became productive quickly using this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

9. The system gives error messages that clearly tell me how to fix problems.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

10. Whenever I make a mistake using the system, I recover easily and quickly.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

11. The help information, such as on-screen messages and other documentation provided with this system is clear.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

12. It is easy to find the information I need.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

13. The information provided with the system is easy to understand.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

14. The information is effective in helping me report a crime.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

15. The organization of information on the system screens is clear.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

16. I like using the interface of this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

17. This system has all the functions and capabilities I expect it to have.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

18. Overall, I am satisfied with this system.

STRONGLY AGREE 1 2 3 4 5 6 7 STRONGLY DISAGREE

Questionnaire on User Preferences

Preamble

I humbly invite you to participate in this research. I am a PhD student at the University of Cape Town. My research seeks to understand users' privacy preferences. A user's privacy preference is the ability of the user to choose among three different privacy levels: low, neutral and high. A high privacy level means you are extremely privacy-conscious. A low privacy level means you are simply privacy-conscious. A neutral privacy level means you are neither extremely nor marginally privacy-conscious.

Anybody is eligible to complete this questionnaire. However, our focus is on people who have been victims of crime such as theft, fraud, family violence, etc. The questionnaire should only take 5-10

minutes to complete. Your co-operation in completing this questionnaire by responding to the following questions would be greatly appreciated. To aid in ensuring anonymity, please do not put your name on the questionnaire. If at any time you feel uncomfortable with your participation in this research, you will be at liberty to opt out. Your participation is completely voluntary.

Biographical Details

1. I am a: Female Male
2. My age is

Crime Details

3. Have you ever been a victim of any type of crime (e.g. theft, fraud, family violence etc)? Yes No
4. If no, please proceed to question 11 Yes No
5. If yes, please describe the crime briefly?
6. Did you report the crime to the police? Yes No
7. If yes, how did you feel after reporting the crime to the police? If no please proceed to question 8.

Privacy and Data Sharing

8. As a follow-up on question 3, would you like the police to share your information with a trusted third party for research or analysis purposes? Yes No
9. If no, please state your reason.
10. If yes, please choose your privacy preference or your privacy sensitivity. A high privacy preference implies that your data might be less significant to the third party and a low privacy preference implies that your data might be more significant. N.B.: If the police share your information with a trusted third party, all the information that can uniquely identify you (such as your name, ID Number, telephone Number, email address) will be removed irrespective of your privacy preference.

11. If you were to be a victim of any crime, would you like to share your data with a trusted third party for research or analysis purpose? If yes, please choose your privacy preference? A high privacy preference implies that your data might be less significant to the third party and a low privacy preference implies your data might be more significant. N.B.: If the police share your information with a trusted third party, all the information that can uniquely identify you (such as your name, ID Nos, Telephone No, Email Address) will be removed irrespective of your privacy preference.

12. In detail, please provide any other comment(s).

A.3 Extract of Responses Obtained from User Privacy Preferences Survey

Serial number	Privacy preference level	Gender	Age group	Present educational level / Occupation
01	Neutral	Male	31-35	PhD
02	Low	Female	26-30	PhD
03	Neutral	Female	26-30	PhD
04	Neutral	Female	26-30	PhD
05	Low	Male	26-30	PhD
06	High	Male	31-35	PhD
07	Low	Female	31-35	PhD
08	High	Female	31-35	PhD
09	Neutral	Male	36-40	PhD
10	Low	Male	26-30	Msc
11	Low	Male	26-30	PhD
12	Neutral	Male	26-30	PhD
13	Neutral	Male	26-30	PhD
14	Neutral	Male	26-30	PhD
15	High	Male	31-35	Msc
16	High	Female	31-35	PhD
17	Neutral	Male	26-30	Msc
18	High	Male	26-30	Msc
19	Low	Male	36-40	Mtech
20	Neutral	Male	31-35	PhD
21	Low	Male	31-35	Msc
22	Low	Male	36-40	Msc
23	High	Male	36-40	PhD
24	High	Male	31-35	PhD

Serial number	Victim of crime?	Crime experienced	Highest educational qualification
01	Yes	Fraud	Msc
02	Yes	Armed Robbery	Msc
03	No	None	Msc
04	Yes	Theft (Phone,money)	Msc
05	Yes	Theft/Bulgary	Msc
06	Yes	Theft	Msc
07	Yes	Theft	Msc
08	Yes	Theft (Phone)	Msc
09	Yes	Mugging	Msc
10	Yes	Robbery	Bsc
11	Yes	Theft	Msc
12	Yes	Theft	Msc
13	No		Msc
14	Yes	Assault	Msc
15	No		Bsc (Hons)
16	No		Msc
17	Yes	Bulgary	Honours
18	Yes	Vehicle Theft	Honours
19	Yes	Theft	HND/Btech
20	Yes	Theft	Msc
21	Yes	Theft	Bsc
22	Yes	Theft	B.Eng
23	Yes	Car Snatching	Msc
24	Yes	Bulgary	Msc

Serial number	Share Data with 3rd Party	Reason for choice of privacy
01	Yes	Personality, to help people
02		To help reduce crime
03		Gravity of crime will determine
04		This crime is not sensitive
05		This crime is not sensitive
06		Personality
07		Since her personal data is removed
08	No	Personality
09		
10		
11		He has nothing to hide, personality
12		To help research, crime is not too sensitive
13		
14		To help reduce crime
15		
16	No	Cultural Background, personality
17		Help in the investigation
18	Yes/No	Personality
19		To help reduce crime/research
20		Personality
21		Crime is not sensitive
22		Residential address of QI should be generalized but other QI can be left
23		
24		Personality

Appendix B

Result Overview

B.1 Raw Data from the Evaluation of CryHelp App

B.1.1 Overview of the Different Parameters Evaluated in the CryHelp App

					Assistance	Assistance						
	Participant	1	2	3	4	5	6	7	8	9	10	Average	Standard Deviation
Question													
1		2	2	1	3	3	1	2	3	2	3	2.2	0.788810638
2		3	2	1	2	3	3	3	2	1	2	2.2	0.788810638
3		2	2	2	2	3	2	2	3	2	3	2.3	0.483045892
4		4	3	4	2	3	3	4	2	4	3	3.2	0.788810638
5		2	1	2	2	2	1	1	2	2	2	1.7	0.483045892
6		3	2	3	2	4	4	3	3	2	1	2.7	0.948683298
7		3	2	2	1	3	2	1	2	2	3	2.1	0.737864787
8		4	1	4	2	2	2	3	3	4	3	2.8	1.032795559
9		4	0	4	4	0	0	0	4	4	4	4	0
10		4	4	2	2	0	2	2	0	2	0	2.571428571	0.975900073
11		2	0	2	1	2	2	1	2	2	1	1.5	0.5
12		2	2	4	1	2	2	2	3	1	2	2.1	0.875595036
13		2	2	2	2	2	3	3	1	2	2	2.1	0.567646212
14		2	1	4	1	3	2	3	4	2	2	2.4	1.0749677
15		1	2	2	2	3	3	2	1	1	3	2	0.816496581
16		1	1	4	2	2	4	1	1	2	2	2	1.154700538
17		2	1	5	3	3	4	4	2	3	3	3	1.154700538
18		1	2	3	2	2	3	2	2	1	1	1.9	0.737864787

Figure B.1: Users' Evaluation of Different Parameters of the CryHelp App. some of the parameters measured are, interface quality, simplicity, user's satisfaction, productivity, security, clarity, effectiveness, efficiency, resilience.

B.1.2 Data Showing Time Taken for each Task and Degree of Easiness of Completing the Task

		Ease of Use				Time			
	Task	1	2	3	4	1	2	3	4
Participant									
1		2	1	0	1	2	1	0	1
2		2	2	1	0	1	2	1	0
3		1	0	2	0	2	0	2	0
4		2	5	1	2	2	5	1	2
5		3	2	1	0	2	1	1	0
6		2	3	2	1	2	3	2	1
7		2	3	1	2	2	2	2	2
8		3	2	1	0	2	2	1	0
9		2	4	4	1	2	3	4	1
10		1	2	2	0	1	2	2	0
Average		2	2.666666667	1.666666667	1.4	1.8	2.333333	1.777778	1.4

Figure B.2: Data from the Evaluation of CryHelp App with Focus on the Ease and Time Taken to Complete the Four Major Tasks. The tasks are full crime report, taking an image, tagging an image and inputting gesture.

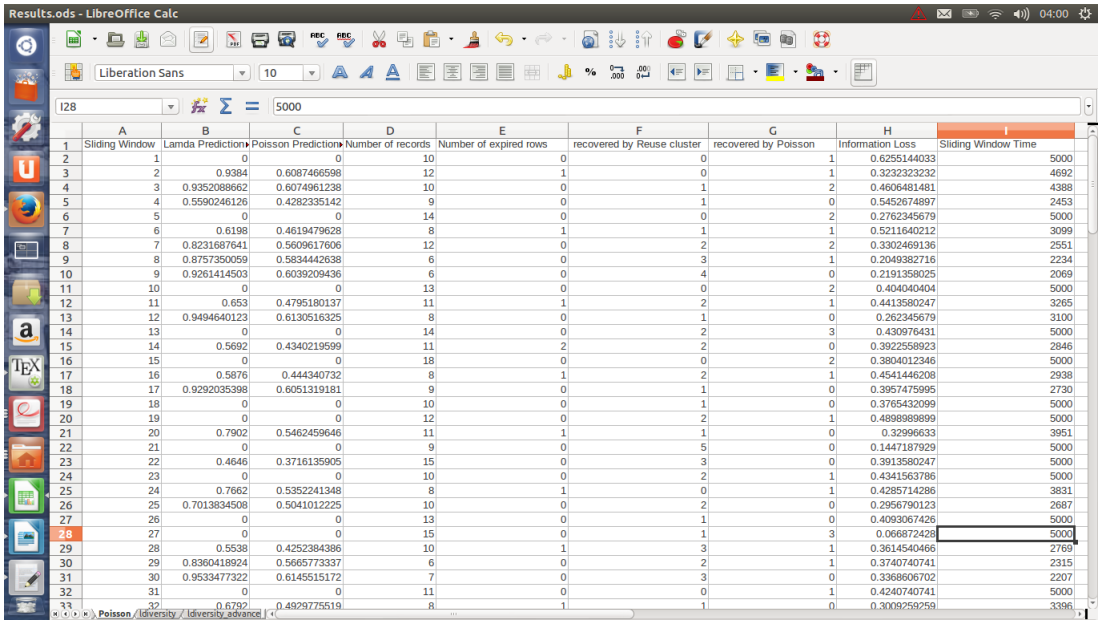
B.1.3 Data Showing Overall Time Taken to Complete and Send the Report

Participant	Overall Time (milleseconds)	
1	245934	
2	482914	
3	580443	
4	494243	
5	165948	
6	445789	
7	274220	
8	477980	
9	537164	
10	124550	
		382919
	Average	164181
	Standard Deviation	

Figure B.3: Data from the Evaluation of CryHelp App with Focus the on the Overall Time Taken to Complete and Send the Whole Application.

B.2 Raw Data Obtained from Evaluation of ABRS and the Different Anonymization Schemes

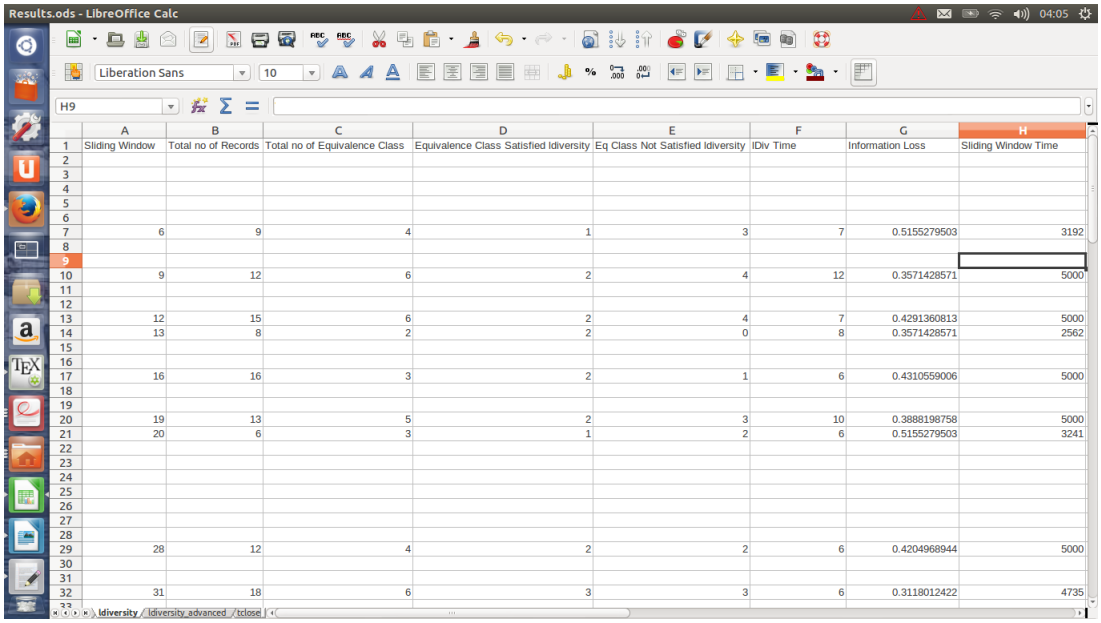
B.2.1 ABRS and K-Anonymity



	A	B	C	D	E	F	G	H	I
1	Sliding Window	Lamda Prediction	Poisson Prediction	Number of records	Number of expired rows	recovered by Reuse cluster	recovered by Poisson	Information Loss	Sliding Window Time
2	1	0	0	10	0	0	1	0.6255144033	5000
3	2	0.9384	0.6087466598	12	1	0	1	0.3232323232	4692
4	3	0.9352088662	0.6074961238	10	0	1	2	0.4606481481	4388
5	4	0.5590246126	0.4282335142	9	0	1	0	0.5452674897	2453
6	5	0	0	14	0	0	2	0.2762345679	5000
7	6	0.6198	0.4619479628	8	1	1	1	0.5211640212	3099
8	7	0.8231687641	0.5609617606	12	0	2	2	0.3302469136	2551
9	8	0.8757350059	0.5834442638	6	0	3	1	0.2049382716	2234
10	9	0.9261414503	0.6039209436	6	0	4	0	0.2191358025	2069
11	10	0	0	13	0	0	2	0.404040404	5000
12	11	0.653	0.4795180137	11	1	2	1	0.4413580247	3265
13	12	0.9494640123	0.6130516325	8	0	1	0	0.262345679	3100
14	13	0	0	14	0	2	3	0.430976431	5000
15	14	0.5692	0.4340219599	11	2	2	0	0.3922558923	2846
16	15	0	0	18	0	0	2	0.3804012346	5000
17	16	0.5876	0.444340732	8	1	2	1	0.4541446208	2938
18	17	0.9292035398	0.6051319181	9	0	1	0	0.3957475995	2730
19	18	0	0	10	0	1	0	0.3765432099	5000
20	19	0	0	12	0	2	1	0.4898989899	5000
21	20	0.7902	0.5462459646	11	1	1	0	0.32996633	3951
22	21	0	0	9	0	5	0	0.1447187929	5000
23	22	0.4646	0.3716135905	15	0	3	0	0.3913580247	5000
24	23	0	0	10	0	2	1	0.4341563786	5000
25	24	0.7662	0.5352241348	8	1	0	1	0.4285714286	3831
26	25	0.7013834508	0.5041012225	10	0	2	0	0.2956790123	2687
27	26	0	0	13	0	1	0	0.4093067426	5000
28	27	0	0	15	0	1	3	0.066872428	5000
29	28	0.5538	0.4252384386	10	1	3	1	0.3614540466	2769
30	29	0.8360418924	0.5665773337	6	0	2	1	0.3740740741	2315
31	30	0.9533477322	0.6145515172	7	0	3	0	0.3368606702	2207
32	31	0	0	11	0	0	1	0.4240740741	5000
33	32	0.6792	0.4929775519	8	1	1	0	0.3009259259	3396

Figure B.4: ABRS performance on Dataset 1 where k = 2, prob. Threshold = 0.4. That is, each equivalence class or cluster of the sliding window under consideration requires at least two records before anonymity can be ensured. For cases where the minimum k-privacy threshold is not met, the ABRS using the Poisson model determines the probability that anonymity can be guaranteed in the next sliding window in such a manner that privacy is preserved and record expiration is minimal.

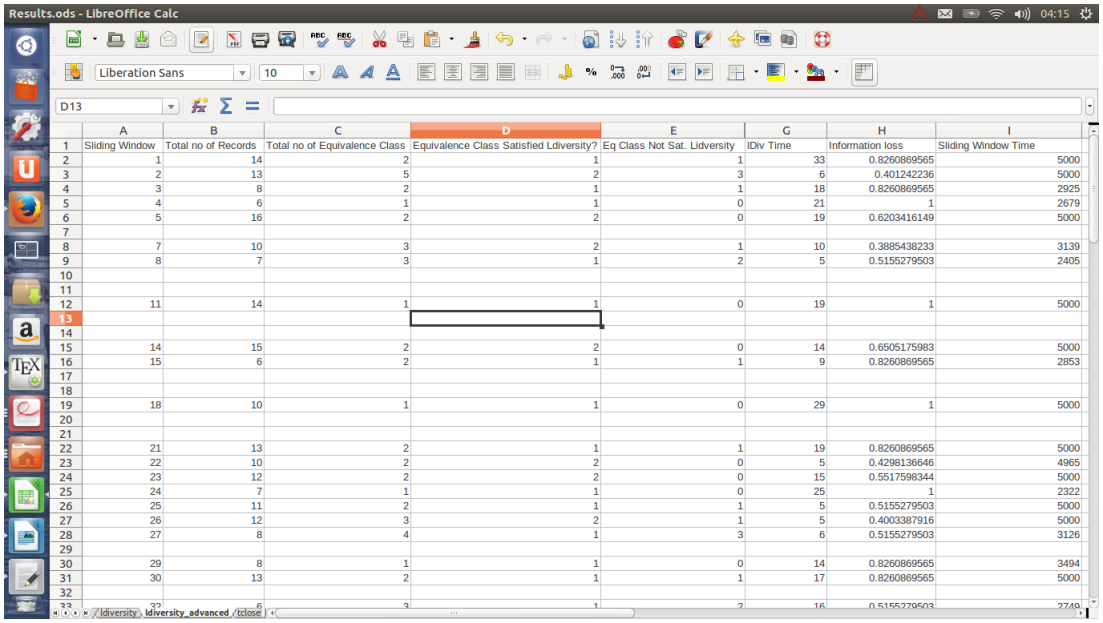
B.2.2 ABRS and Basic ℓ-diversity



	A	B	C	D	E	F	G	H
1	Sliding Window	Total no of Records	Total no of Equivalence Class	Equivalence Class Satisfied ldiversity	Eq Class Not Satisfied ldiversity	lDiv Time	Information Loss	Sliding Window Time
7	6	9	4	1	3	7	0.5155279503	3192
10	9	12	6	2	4	12	0.3571428571	5000
13	12	15	6	2	4	7	0.4291360813	5000
14	13	8	2	2	0	8	0.3571428571	2562
17	16	16	3	2	1	6	0.4310559006	5000
20	19	13	5	2	3	10	0.3888198758	5000
21	20	6	3	1	2	6	0.5155279503	3241
29	28	12	4	2	2	6	0.4204968944	5000
32	31	18	6	3	3	6	0.3118012422	4735

Figure B.5: Performance of ABRS and Basic ℓ-Diversity for Dataset 1 where k $= 2$, $\ell = 3$, $\alpha = 0.1$ and prob. Threshold $= 0.4$. That is, each equivalence class or cluster of the sliding window under consideration requires at least two records and three distinct sensitive values before anonymity can be ensured. The blank spaces represent instances where ℓ-diversity was not satisfied after k-anonymity and those instantiated advanced diversity.

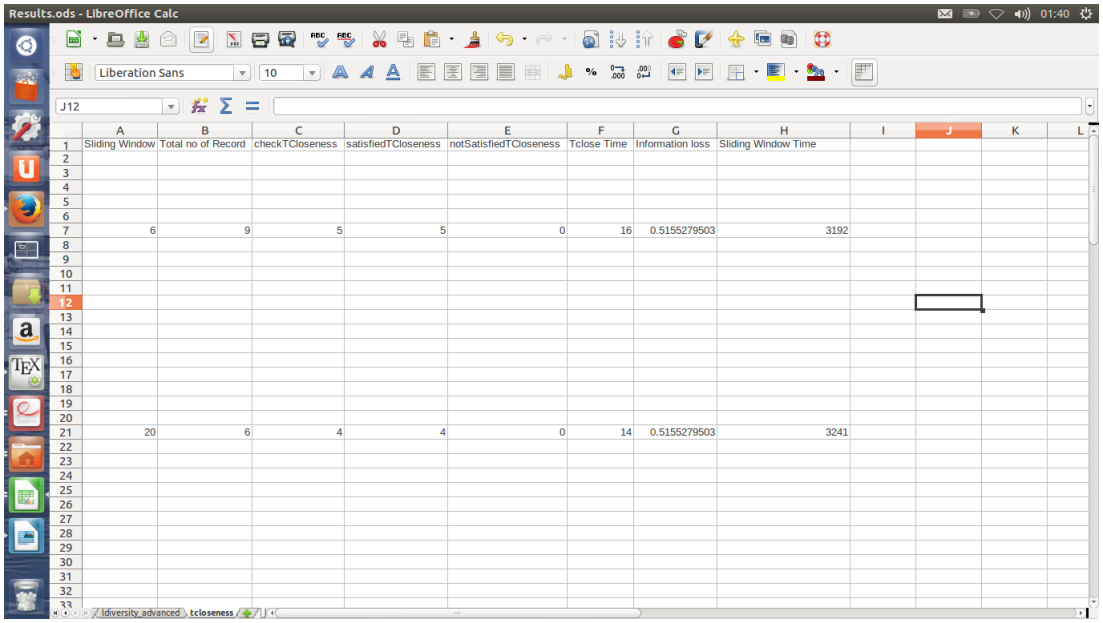
B.2.3 ABRS and Advanced ℓ-diversity



	A	B	C	D	E	G	H	I
1	Sliding Window	Total no of Records	Total no of Equivalence Class	Equivalence Class Satisfied Ldiversity?	Eq Class Not Sat. Ldiversity	lDiv Time	Information loss	Sliding Window Time
2	1	14	2	1	1	33	0.8260869565	5000
3	2	13	5	2	3	6	0.401242236	5000
4	3	8	2	1	1	18	0.8260869565	2925
5	4	6	1	1	0	21	1	2679
6	5	16	2	2	0	19	0.6203416149	5000
7								
8	7	10	3	2	1	10	0.3885438233	3139
9	8	7	3	1	2	5	0.5155279503	2405
10								
11								
12	11	14	1	1	0	19	1	5000
13								
14								
15	14	15	2	2	0	14	0.6505175983	5000
16	15	6	2	1	1	9	0.8260869565	2853
17								
18								
19	18	10	1	1	0	29	1	5000
20								
21								
22	21	13	2	1	1	19	0.8260869565	5000
23	22	10	2	2	0	5	0.4298136646	4965
24	23	12	2	2	0	15	0.5517598344	5000
25	24	7	1	1	0	25	1	2322
26	25	11	2	1	1	5	0.5155279503	5000
27	26	12	3	2	1	5	0.4003387916	5000
28	27	8	4	1	3	6	0.5155279503	3126
29								
30	29	8	1	1	0	14	0.8260869565	3494
31	30	13	2	1	1	17	0.8260869565	5000
32								
33	32	6	3	1	2	16	0.5155279503	2749

Figure B.6: Performance of ABRS and Advanced ℓ-Diversity for Dataset 1 where k $= 2$, $\ell = 3$, $\alpha = 0.1$ and prob. Threshold $= 0.4$. This was needed for cases where basic ℓ-diversity was not satisfied.

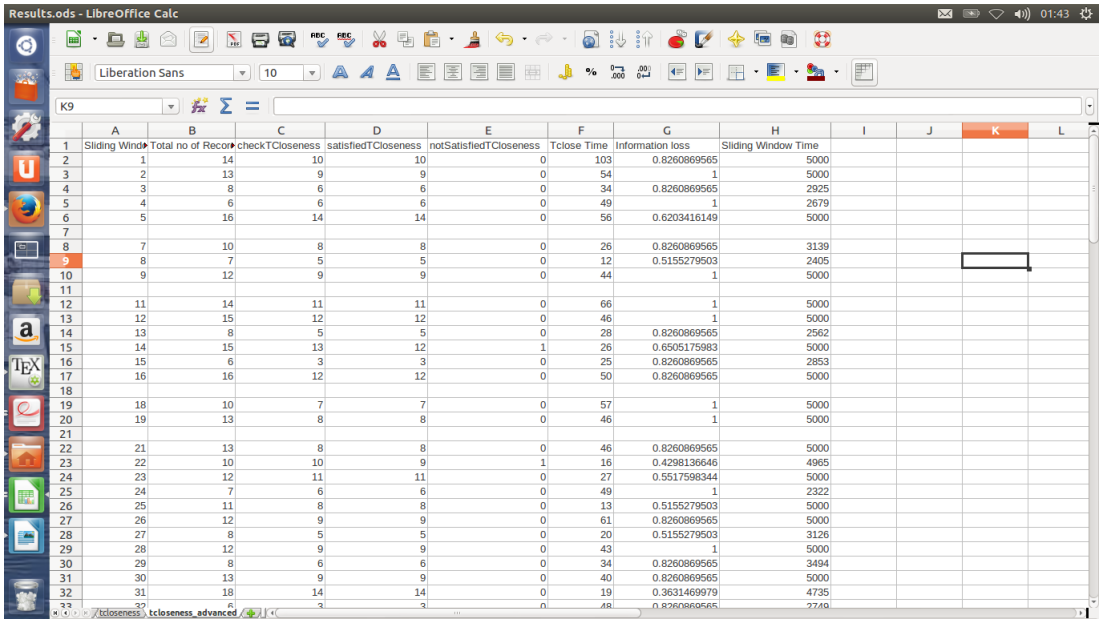
B.2.4 ABRS and Basic t-Closeness



Sliding Window	Total no of Record	checkTCloseness	satisfiedTCloseness	notSatisfiedTCloseness	Tclose Time	Information loss	Sliding Window Time
6	9	5	5	0	16	0.5155279503	3192
20	6	4	4	0	14	0.5155279503	3241

Figure B.7: Performance of ABRS and Basic t-Closeness for Dataset 1 where k = 2, $t = 0.15$, $\beta = 0.1$ and prob. threshold = 0.4. That is, each equivalence class or cluster of the sliding window under consideration requires at least two records and the difference between the distribution of sensitive values in the data stream and cluster has to be at most 0.15 before anonymity can be ensured. The blank spaces represent instances where t-closeness was not satisfied after k-anonymity and those instantiated advanced t-closeness.

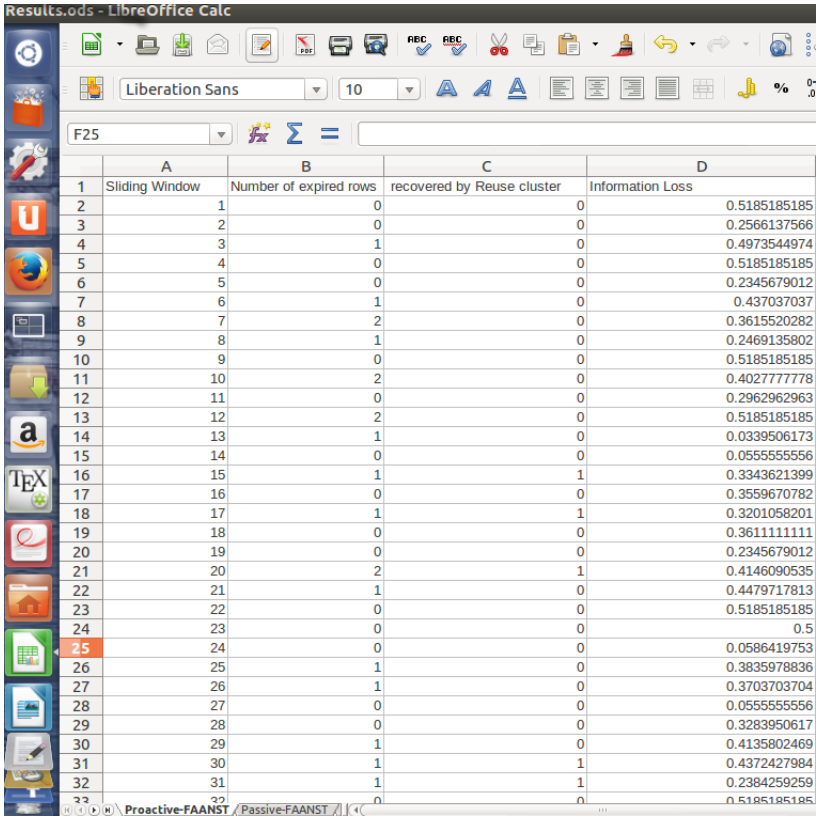
B.2.5 ABRS and Advanced t-Closeness



	A	B	C	D	E	F	G	H
1	Sliding Windo	Total no of Recor	checkTCloseness	satisfiedTCloseness	notSatisfiedTCloseness	Tclose Time	Information loss	Sliding Window Time
2	1	14	10	10	0	103	0.8260869565	5000
3	2	13	9	9	0	54	1	5000
4	3	8	6	6	0	34	0.8260869565	2925
5	4	6	6	6	0	49	1	2679
6	5	16	14	14	0	56	0.6203416149	5000
7								
8	7	10	8	8	0	26	0.8260869565	3139
9	8	7	5	5	0	12	0.5155279503	2405
10	9	12	9	9	0	44	1	5000
11								
12	11	14	11	11	0	66	1	5000
13	12	15	12	12	0	46	1	5000
14	13	8	5	5	0	28	0.8260869565	2562
15	14	15	13	12	1	26	0.6505175983	5000
16	15	6	3	3	0	25	0.8260869565	2853
17	16	16	12	12	0	50	0.8260869565	5000
18								
19	18	10	7	7	0	57	1	5000
20	19	13	8	8	0	46	1	5000
21								
22	21	13	8	8	0	46	0.8260869565	5000
23	22	10	10	9	1	16	0.4298136646	4965
24	23	12	11	11	0	27	0.5517598344	5000
25	24	7	6	6	0	49	1	2322
26	25	11	8	8	0	13	0.5155279503	5000
27	26	12	9	9	0	61	0.8260869565	5000
28	27	8	5	5	0	20	0.5155279503	3126
29	28	12	9	9	0	43	1	5000
30	29	8	6	6	0	34	0.8260869565	3494
31	30	13	9	9	0	40	0.8260869565	5000
32	31	18	14	14	0	19	0.3631469979	4735
33	32	6	3	3	0	48	0.8260869565	2749

Figure B.8: Performance of ABRS and Advanced t-Closeness for Dataset 1 where k = 2, $t = 0.15$, $\beta = 0.1$ and prob. Threshold = 0.4. This was needed for cases where basic t-closeness was not satisfied.

B.2.6 Sample of Raw Data Obtained from Experiments on Proactive-FAANST



	A	B	C	D
1	Sliding Window	Number of expired rows	recovered by Reuse cluster	Information Loss
2	1	0	0	0.5185185185
3	2	0	0	0.2566137566
4	3	1	0	0.4973544974
5	4	0	0	0.5185185185
6	5	0	0	0.2345679012
7	6	1	0	0.437037037
8	7	2	0	0.3615520282
9	8	1	0	0.2469135802
10	9	0	0	0.5185185185
11	10	2	0	0.4027777778
12	11	0	0	0.2962962963
13	12	2	0	0.5185185185
14	13	1	0	0.0339506173
15	14	0	0	0.0555555556
16	15	1	1	0.3343621399
17	16	0	0	0.3559670782
18	17	1	1	0.3201058201
19	18	0	0	0.3611111111
20	19	0	0	0.2345679012
21	20	2	1	0.4146090535
22	21	1	0	0.4479717813
23	22	0	0	0.5185185185
24	23	0	0	0.5
25	24	0	0	0.0586419753
26	25	1	0	0.3835978836
27	26	1	0	0.3703703704
28	27	0	0	0.0555555556
29	28	0	0	0.3283950617
30	29	1	0	0.4135802469
31	30	1	1	0.4372427984
32	31	1	1	0.2384259259
33	32	0	0	0.5185185185

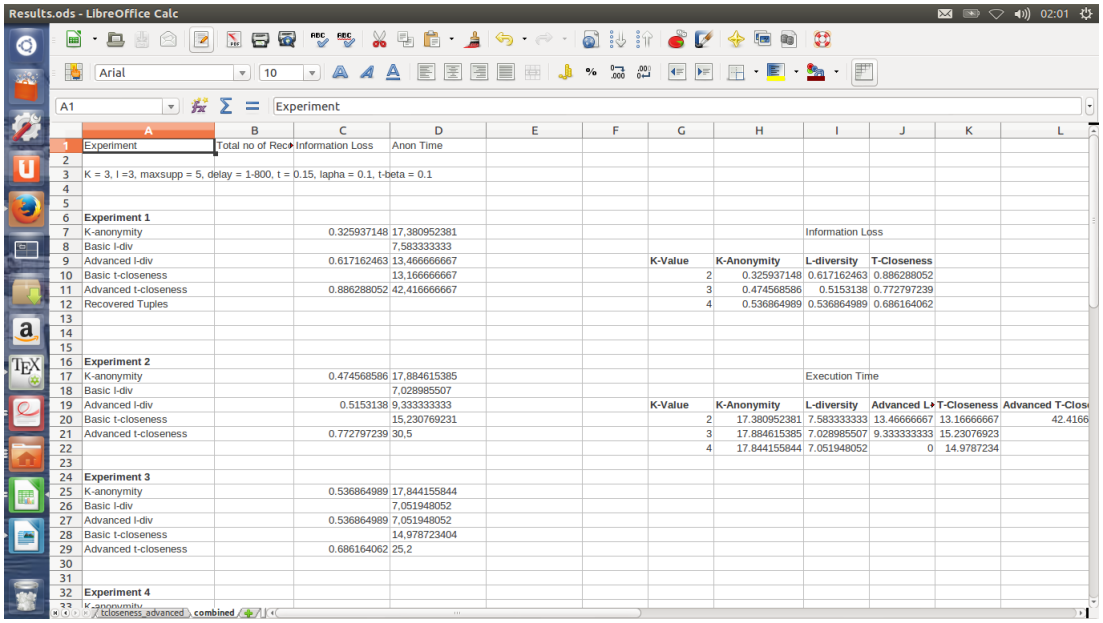
Figure B.9: Performance of Proactive-FAANST for Dataset 1 where k = 2. This was needed for comparison with ABRS.

B.2.7 Sample of Raw Data Obtained from Experiments on Passive-FAANST

	Sliding Window	Number of expired rows	Number of Suppressed Records	recovered by Reuse cluste	Information Loss
2	1	0	0	0	0.4970414201
3	2	1	0	0	0.2527472527
4	3	1	1	0	0.482671175
5	4	0	0	0	0.4970414201
6	5	0	0	0	0.224852071
7	6	0	0	0	0.2958579882
8	7	2	1	0	0.3068469992
9	8	1	0	0	0.1337278107
10	9	0	0	0	0.3360946746
11	10	1	1	0	0.4769230769
12	11	0	0	0	0.2958579882
13	12	2	2	0	0.5088757396
14	13	1	0	0	0.0394477318
15	14	0	0	0	0.2406311637
16	15	1	1	0	0.5808678501
17	16	0	1	0	0.3964497041
18	17	0	0	0	0.3461538462
19	18	1	1	0	0.4887573964
20	19	0	0	0	0.1153846154
21	20	2	2	0	0.6646942801
22	21	0	0	0	0.4970414201
23	22	1	0	0	0.5
24	23	0	0	0	0.0562130178
25	24	1	0	0	0.3677092139
26	25	1	0	0	0.3629191321
27	26	0	0	0	0.0532544379
28	27	0	0	0	0.3147928994
29	28	1	0	0	0.3964497041
30	29	2	1	1	0.5240912933
31	30	0	0	0	0.2699704142
32	31	0	0	0	0.4970414201

Figure B.10: Performance of Passive-FAANST for Dataset 1 where k = 2. This was needed for comparison with ABRS.

B.2.8 Sample of Tabular Representation of Experiment Summary



Results.ods - LibreOffice Calc

	A	B	C	D
1	Experiment	Total no of Rec	Information Loss	Anon Time
2				
3	K = 3, l =3, maxsupp = 5, delay = 1-800, t = 0.15, lapha = 0.1, t-beta = 0.1			
4				
5				
6	**Experiment 1**			
7	K-anonymity		0.325937148	17,380952381
8	Basic l-div			7,583333333
9	Advanced l-div		0.617162463	13,466666667
10	Basic t-closeness			13,166666667
11	Advanced t-closeness		0.886288052	42,416666667
12	Recovered Tuples			
13				
14				
15				
16	**Experiment 2**			
17	K-anonymity		0.474568586	17,884615385
18	Basic l-div			7,028985507
19	Advanced l-div		0.5153138	9,333333333
20	Basic t-closeness			15,230769231
21	Advanced t-closeness		0.772797239	30,5
22				
23				
24	**Experiment 3**			
25	K-anonymity		0.536864989	17,844155844
26	Basic l-div			7,051948052
27	Advanced l-div		0.536864989	7,051948052
28	Basic t-closeness			14,978723404
29	Advanced t-closeness		0.686164062	25,2
30				
31				
32	**Experiment 4**			

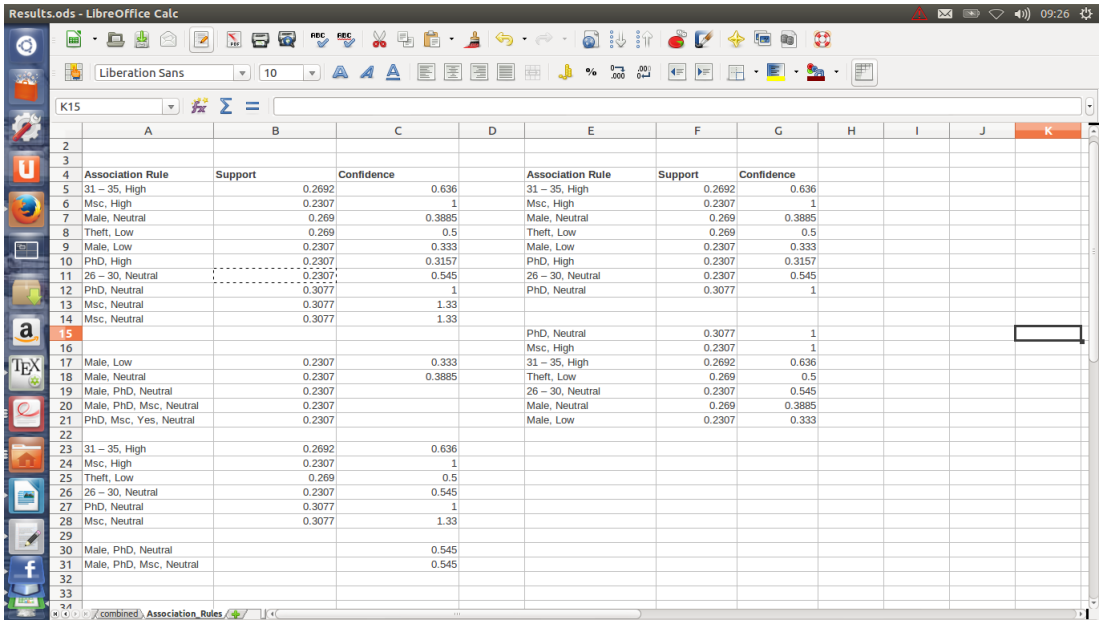
	G	H	I	J	K	L
1						
2						
3						
4						
5						
6						
7			Information Loss			
8						
9	**K-Value**	**K-Anonymity**	**L-diversity**	**T-Closeness**		
10	2	0.325937148	0.617162463	0.886288052		
11	3	0.474568586	0.5153138	0.772797239		
12	4	0.536864989	0.536864989	0.686164062		
13						
14						
15						
16						
17			Execution Time			
18						
19	**K-Value**	**K-Anonymity**	**L-diversity**	**Advanced L**	**T-Closeness**	**Advanced T-Clos**
20	2	17.380952381	7.583333333	13.46666667	13.16666667	42.4166
21	3	17.884615385	7.028985507	9.333333333	15.23076923	
22	4	17.844155844	7.051948052	0	14.9787234	
23						
24						
25						
26						
27						
28						
29						
30						
31						
32						

closeness_advanced | combined

Figure B.11: Overview of Different Experimental Results. For the sake of precision, a maximum of ten runs for each of the experiments were conducted.

B.3 User Privacy Preferences

B.3.1 Sample of Various Association Rules Generated from the Survey



	A	B	C	D	E	F	G
4	**Association Rule**	**Support**	**Confidence**		**Association Rule**	**Support**	**Confidence**
5	31 – 35, High	0.2692	0.636		31 – 35, High	0.2692	0.636
6	Msc, High	0.2307	1		Msc, High	0.2307	1
7	Male, Neutral	0.269	0.3885		Male, Neutral	0.269	0.3885
8	Theft, Low	0.269	0.5		Theft, Low	0.269	0.5
9	Male, Low	0.2307	0.333		Male, Low	0.2307	0.333
10	PhD, High	0.2307	0.3157		PhD, High	0.2307	0.3157
11	26 – 30, Neutral	0.2307	0.545		26 – 30, Neutral	0.2307	0.545
12	PhD, Neutral	0.3077	1		PhD, Neutral	0.3077	1
13	Msc, Neutral	0.3077	1.33				
14	Msc, Neutral	0.3077	1.33				
15					PhD, Neutral	0.3077	1
16					Msc, High	0.2307	1
17	Male, Low	0.2307	0.333		31 – 35, High	0.2692	0.636
18	Male, Neutral	0.2307	0.3885		Theft, Low	0.269	0.5
19	Male, PhD, Neutral	0.2307			26 – 30, Neutral	0.2307	0.545
20	Male, PhD, Msc, Neutral	0.2307			Male, Neutral	0.269	0.3885
21	PhD, Msc, Yes, Neutral	0.2307			Male, Low	0.2307	0.333
22							
23	31 – 35, High	0.2692	0.636				
24	Msc, High	0.2307	1				
25	Theft, Low	0.269	0.5				
26	26 – 30, Neutral	0.2307	0.545				
27	PhD, Neutral	0.3077	1				
28	Msc, Neutral	0.3077	1.33				
29							
30	Male, PhD, Neutral		0.545				
31	Male, PhD, Msc, Neutral		0.545				
32							
33							

Figure B.12: Association Rules for User Privacy Preferences

Bibliography

[1] Statistics: the poisson distribution. `http://www.umass.edu/wsp/resources/poisson/index.html`. Accessed: 2017-07-11.

[2] C. C. Aggarwal and S.Y. Philip. A general survey of privacy-preserving data mining models and algorithms. In *Privacy-Preserving Data Mining: Models and Algorithms*, pages 11–52. Springer, 2008.

[3] A. Agresti. Categorical data analysis. *3rd edn, John Wiley & Sons, Inc.*, 2012.

[4] R.J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *Proceedings of 21st International Conference on Data Engineering (ICDE)*, pages 217–228. IEEE, 2005.

[5] A. Blum, K. Ligett, and A. Rott. A learning theory approach to noninteractive database privacy. *Journal of the ACM (JACM)*, 60(2):12, 2013.

[6] D. Bradfield and L. Underhill. *Introstat (Introduction to Statistics Textbook)*. University of Cape Town, 2014.

[7] M. Burke and Anne V.D.M. Kayem . K-anonymity for privacy preserving crime data publishing in resource constrained environments. In *Proceedings of 28th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pages 833–840. IEEE, 2014.

[8] M.J. Burke. Enabling anonymous crime reporting on mobile phones in the developing world. *Masters Dissertation, University of Cape Town*, 2013.

[9] M. Butler. Android: Changing the mobile landscape. *IEEE Pervasive Computing*, 10(1):4–7, 2011.

[10] J. Cao, B. Carminati, E. Ferrari, and K. L. Tan. Castle: Continuously anonymizing data streams. *Dependable and Secure Computing, IEEE Transactions on*, 8(3):337–352, 2011.

[11] R. Chen, N. Mohammed, B.C. Fung, B.C. Desai, and L.Xion. Publishing set-valued data via differential privacy. *Proceedings of the VLDB Endowment*, 4(11):1087–1098, 2011.

[12] C.C. Chiu and C.Y. Tsai. A k-anonymity clustering method for effective data privacy preservation. *Advanced Data Mining and Applications, Springer*, pages 89–99, 2007.

[13] I. H. Chuang, S.H. Li, K.C. Huang, and Y.H. Kuo. An effective privacy protection scheme for cloud computing. In *Proceedings of 13th International Conference on Advanced communication Technology (ICACT, 2011)*, pages 260–265. IEEE, 2011.

[14] V. Ciriani, S. Foresti, and P. Samarati. Microdata protection. In *Secure data management in decentralized systems*, pages 291–321. Springer, 2007.

[15] L. H. Cox. Suppression methodology and statistical disclosure control. *Journal of the American Statistical Association*, 75(370):377–385, 1980.

[16] M. Crouch and H. McKenzie. The logic of small samples in interview-based qualitative research. *Social science information*, 45(4):483–499, 2006.

[17] Y. Cui, J. Chipchase, and F. Ichikawa. A cross culture study on phone carrying and physical personalization. *Usability and Internationalization. HCI and Culture*, pages 483–492, 2007.

[18] J. Domingo-Ferrer, D. Sánchez, and G. Rufian-Torrell. Anonymization of nominal data based on semantic marginality. *Information Sciences*, 242:35–48, 2013.

[19] J. Domingo-Ferrer and J. Soria-Comas. From t-closeness to differential privacy and vice versa in data anonymization. *Knowledge-Based Systems*, 74:151–158, 2015.

[20] J. Domingo-Ferrer and V. Torra. A critique of k-anonymity and some of its enhancements. In *Proceedings of the 3rd International Conference on Availability, Reliability and Security (ARES)*, pages 990–993. IEEE, 2008.

[21] C. Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer, 2008.

[22] C. Dwork. Differential privacy. In *Encyclopedia of Cryptography and Security*, pages 338–340. Springer, 2011.

[23] C. Dwork, M. Naor, O. Reingold, G.N. Rothblum, and S.Vadhan. On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 381–390. ACM, 2009.

[24] B. Fung, K. Wang, A.W.C. Fu, and J. Pei. Anonymity for continuous data publishing. In *Proceedings of the 11th international conference on Extending database technology: Advances in database technology*, pages 264–275. ACM, 2008.

[25] B. Gedik and L. Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE Transactions on Mobile Computing*, 7(1):1–18, 2008.

[26] A. Gkoulalas-Divanis and G. Loukides. Overview of patient data anonymization. In *Anonymization of Electronic Medical Records to Support Clinical Analysis*, pages 9–30. Springer, 2013.

[27] H. Gomaa and D.B. Scott. Prototyping as a tool in the specification of user requirements. *In Proceedings of the 5th international conference on Software engineering*, pages 333–342, 1981.

[28] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of the 1st international conference on Mobile systems, applications and services*, pages 31–42. ACM, 2003.

[29] K. Guo and Q. Zhang. Fast clustering-based anonymization approaches with time constraints for data streams. *Knowledge-Based Systems, Elsevier*, 46:95–108, 2013.

[30] M. Hardt, K. Ligett, and F. McSherry. A simple and practical algorithm for differentially private data release. In *Advances in Neural Information Processing Systems*, pages 2339–2347, 2012.

[31] H.Chen, W. Chung, J. Xu, G. Wang, Y. Qin, and M. Chau. Crime data mining: A general framework and some examples. *In Journal of IEEE Computer, 37(4)*, pages 50–56, 2004.

[32] Y. He and J.F. Naughton. Anonymization of set-valued data via top-down, local generalization. *Proceedings of the VLDB Endowment*, 2(1):934–945, 2009.

[33] J. Hipp, U. GÃ$\frac{1}{4}$ntzer, and G. Nakhaeizadeh. Algorithms for association rule mining a general survey and comparison. *ACM sigkdd explorations newsletter, 2(1)*, pages 58–64, 2000.

[34] O. Isafiade and A. Bagula. Efficient frequent pattern knowledge for crime situation recognition in developing countries. In *Proceedings of the 4th Annual Symposium on Computing for Development*, page 21. ACM, 2013.

[35] O. E. Isafiade and A. B. Bagula. Citisafe: Adaptive spatial pattern knowledge using fp-growth algorithm for crime situation recognition. In *proceedings of the IEEE International Symposium on Ubiquitous Intelligence and Autonomic Systems (IEEE-UIAS)*, pages 551–556. IEEE, 2013.

[36] O.E. Isafiade and A.B. Bagula. Data mining trends and applications in criminal science and investigations. *Advances in Data Mining and Database Management, IGI Global*, 2016.

[37] R. Issa. Satisfying k-anonymity:new algorithm and empirical evaluation. *Masters Dissertation, Carleton University*, 2009.

[38] V.S. Iyengar. Transforming data to satisfy privacy constraints. *In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 279–288, 2002.

[39] K. Stern J. Rudd and S. Isensee. Low vs. high-fidelity prototyping debate. *interactions, 3(1)*, pages 76–85, 1996.

[40] K. L Jensen, H. N. Iipito, M. U. Onwordi, and S. Mukumbira. Toward an mpolicing solution for namibia: leveraging emerging mobile platforms and crime mapping. In *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, pages 196–205. ACM, 2012.

[41] Kasper L Jensen, Hedvig NK Iipito, Michel U Onwordi, and Sebastian Mukumbira. Toward an mpolicing solution for namibia: leveraging emerging mobile platforms and crime mapping. In *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, pages 196–205. ACM, 2012.

[42] W. Jiang and C. Clifton. A secure distributed framework for achieving k- anonymity. *The VLDB Journal: The International Journal on Very Large Data Bases*, 15(4):316–333, 2006.

[43] M. E. Kabir, H. Wang, and E. Bertino. Efficient systematic clustering method for k-anonymization. *Acta Informatica*, 48(1):51–66, 2011.

[44] A.V.D.M. Kayem, P. Martin, and S.G. Akl. Effective cryptographic key management for outsourced dynamic data sharing environments. In *Proceedings of the 10th Annual Information Security Conference (ISSA 2011)*, pages 1–8. IEEE, 2011.

[45] A.V.D.M. Kayem, C.T. Vester, and C. Meinel. Automated k-anonymization and l-diversity for shared data privacy. In *Proccedings of 27th International Conference on Database and Expert Systems Applications, Springer International Publishing*, pages 105–120, 2016.

[46] S. Kim, M. K. Sung, and Y. D. Chung. A framework to preserve the privacy of electronic health data streams. *Journal of biomedical informatics*, 50:95–106, 2014.

[47] S. Kumar and P. Phrommathed. Research methodology. In *New Product Development*, pages 43–50. Springer, 2005.

[48] J.R. Lewis. Ibm computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1):57–78, 1995.

[49] F. Li, J. Sun, S. Papadimitriou, G.A. Mihaila, and I. Stanoi. Hiding in the crowd: Privacy preservation on evolving streams through correlation tracking. In *Proceedings of 23rd International Conference on Data Engineering*, pages 686–695. IEEE, 2007.

[50] J. Li, B.C. Ooi, and W. Wang. Anonymizing streaming data for privacy protection. In *Proceedings of the 24th International Conference on Data Engineering*, pages 1367–1369. IEEE, 2008.

[51] N. Li, W. H. Qardaji, and D. Su. Provably private data anonymization: Or, k-anonymity meets differential privacy. *Arxiv preprint*, 2011.

[52] N. Li, T.Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Proceedings of the 23rd International Conference on Data Engineering*, pages 106–115. IEEE, 2007.

[53] S. Li. Poisson process with fuzzy rates. *Fuzzy Optimization and Decision Making*, 9(3):289–305, 2010.

[54] T. Li and N. Li. Towards optimal k-anonymization. *Data & Knowledge Engineering*, 65(1):22–39, 2008.

[55] T. Li and N. Li. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 517–526. ACM, 2009.

[56] L. Lu and X. Ye. An improved weighted-feature clustering algorithm for k-anonymity. In *Proceedings of the 5th International Conference on Information Assurance and Security*, pages 415–418. IEEE, 2009.

[57] A. Machanavajjhala, D. Kifer, and G. Johannes. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[58] A.J. Menezes, P.C. Van Oorschot, and S.A. Vanstone. *Handbook of applied cryptography*. CRC press, 1996.

[59] E. Mohammadian, M. Noferesti, and R. Jalili. Fast: fast anonymization of big data streams. In *Proceedings of the 2014 International Conference on Big Data Science and Computing*, page 23. ACM, 2014.

[60] M. F. Mokbel, C.Y. Chow, and W. G. Aref. The new casper: Query processing for location services without compromising privacy. In *Proceedings of the 32nd international conference on Very large data bases*, pages 763–774. VLDB Endowment, 2006.

[61] M. L. Murphy. *The busy coder's guide to Android development*. United States: CommonsWare., 2008.

[62] M. Mvurya and A. Mbogho. Data-driven intervention-level prediction modeling for academic performance. *Proceedings of the Seventh International Conference on Information and Communication Technologies and Development*, page 2, 2015.

[63] M. Mvurya and A. Mbogho. *Investigating prediction modelling of academic performance for students in rural schools in Kenya*. PhD thesis, University of Cape Town, 2016.

[64] J. Nielsen. Usability engineering. 1994.

[65] J. Nielsen. Usability 101: Introduction to usability. `http://https://www.nngroup.com/articles/usability-101-introduction-to-usability/`, 2003. Accessed: 2017-08-02.

[66] J. Nielsen. How many test users in a usability study. *Nielsen Norman Group*, 4(06), 2012.

[67] K. Patroumpas and T. Sellis. Window specification over data streams. *Current Trends in Database Technology–EDBT 2006*, pages 445–464, 2006.

[68] L. Qiu, Y. Li, X., and Wu. Protecting business intelligence and customer privacy while outsourcing data mining tasks. *Knowledge and Information Systems*, 17(1):99–120, 2008.

[69] Emergency Report. E9 speed dial 9. `http://www.emergency9.co.za/`, 2017. Accessed: 2017-08-02.

[70] A. B. Sakpere and Anne V.D.M. Kayem. A state-of-the-art review of data stream anonymization schemes. In *Information Security in Diverse Computing Environments*, pages 24–50. IGI Global, 2014.

[71] A.B. Sakpere and Anne V.D.M. Kayem. Adaptive buffer resizing for efficient anonymization of streaming data with minimal information loss. In *Proceedings of 1st International Conference on Information Systems Security and Privacy (ICISSP)*, pages 1–11. IEEE, 2015.

[72] A.B. Sakpere and Anne V.D.M. Kayem. Supporting streaming data anonymization with expressions of user privacy preferences. In *Information Systems Security and Privacy. Communications in Computer and Information Science*, volume 576, pages 122–136. Springer, 2015.

[73] A.B. Sakpere, Anne V.D.M. Kayem, and T. Ndlovu. A usable and secure crime reporting system for technology resource constrained context. In *Proceedings of 29th International Conference on Advanced Information Networking and Applications Workshops*, pages 424–429. IEEE, 2015.

[74] P. Samarati. Protecting respondents identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[75] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki. Caravan: Providing location privacy for vanet. Technical report, Department Of Electrical Engineering, Washington University, Seattle, 2005.

[76] R. Sefelin, M. Tscheligi, and V. Giller. Paper prototyping-what is it good for?: a comparison of paper-and computer-based low-fidelity prototyping. In *CHI'03 Extended Abstracts on Human factors in computing systems*, pages 778–779. ACM, 2003.

[77] A. Silberschatz, H. F. Korth, and S. Sudarshan. *Database System Concepts*, volume 4. McGraw-Hill New York, 1997.

[78] A. Skrondal. Multilevel logistic regression for polytomous data and rankings. *Psychometrika 68(2)*, pages 267–287, 2003.

[79] J. Soria-Comas and J. Domingo-Ferrer. Differential privacy via t-closeness in data publishing. In *Proceedings of the 11th Annual Conference on Privacy, Security and Trust (PST)*, pages 27–35. IEEE, 2013.

[80] X. Sun, H. Wang, J. Li, and T.M. Truta. Enhanced p-sensitive k-anonymity models for privacy preserving data publishing. *Transactions on Data Privacy*, 1(2):53–66, 2008.

[81] L. Sweeney. Guaranteeing anonymity when sharing medical data, the datafly system. In *Proceedings of the AMIA Annual Fall Symposium*, page 51. American Medical Informatics Association, 1997.

[82] L. Sweeney. Uniqueness of simple demographics in the us population. Technical report, Carnegie Mellon University, 2000.

[83] L. Sweeney. *Computational disclosure control: a primer on data privacy protection*. PhD thesis, Massachusetts Institute of Technology, 2001.

[84] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):571–588, 2002.

[85] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10(05)*, pages 557–570, 2002.

[86] K. W. Tan, Y. Lin, and K. Mouratidis. Spatial cloaking revisited: Distinguishing information leakage from anonymity. In *International Symposium on Spatial and Temporal Databases*, pages 117–134. Springer, 2009.

[87] T. Tassa and E. Gudes. Secure distributed computation of anonymized views of shared databases. *ACM Transactions on Database System (TODS)*, 37(2):11, 2012.

[88] T.Hornyak. Android grabs record 85 percent smartphone share. pcworld. `http://www.pcworld.com/article/2460020/android-grabs-record-85-percent-smartphone-share.html`. Accessed: 2017-05-20.

[89] B.K Tripathy. Database anonymization techniques with focus on uncertainty and multi-sensitive attributes. In *Handbook of Research on Computational Intelligence for Engineering, Science, and Business*, pages 364–383. IGI Global, 2013.

[90] J.S. Valacich, J.F. George, and J.A. Hoffer. *Essentials of Systems Analysis and Design*. Pearson, 2012.

[91] S.D.C.D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Encryption policies for regulating access to outsourced data. *ACM Transactions on Database Systems (TODS)*, 35(2):12, 2010.

[92] M. Walker, L. Takayama, and J.A. Landay. High-fidelity or low-fidelity, paper or computer? choosing attributes when testing web prototypes. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 46, pages 661–665, 2002.

[93] P. Wang, J. Lu, L. Zhao, and J. Yang. B-castle: An efficient publishing algorithm for k-anonymizing data streams. In *Proceedings of the 2010 Second WRI Global Congress on Intelligent Systems (GCIS)*, volume 2, pages 132–136. IEEE, 2010.

[94] W. Wang, J. Li, C. Ai, and Y. Li. Privacy protection on sliding window of data streams. In *Proceedings of International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 213–221. IEEE, 2007.

[95] R.L. Wilson and P.A. Rosen. Protecting data through perturbation techniques: The impact on knowledge discovery in databases. In *Information Security and Ethics: Concepts, Methodologies, Tools and Applications*, pages 1550–1561. IGI Global, 2008.

[96] M. Wu and X. Ye. Towards the diversity of sensitive attributes in k-anonymity. In *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pages 98–104. IEEE Computer Society, 2006.

[97] X. Xiao and Y. Tao. Personalized privacy preservation. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2006.

[98] T. Xu and Y. Cai. Feeling-based location privacy protection for location-based services. In *Proceedings of the 16th conference on Computer and Communications Security*, pages 348–357. ACM, 2009.

[99] L.C. Mingxuan Yuan and S.Y. Philip. Personalized privacy protection in social networks. In *Proceedings of 37th International Conference on Very Large Database*, volume 4, pages 141–150. VLDB Endowment, 2011.

[100] H. Zakerzadeh and S.L. Osborn. Fast anonymizing algorithm for numerical streaming data. In *Proceedings of 5th International Workshop on Data Privacy Management and 3rd International Conference on Autonomous Spontaneous Security*, pages 36–50, 2011.

[101] H. Zakerzadeh and S.L. Osborn. Delay-sensitive approaches for anonymizing numerical streaming data. *International Journal of Information Security*, pages 1–15, 2013.

[102] J. Zhang, J. Yang, J.Zhang, and Y. Yuan. Kids: K-anonymization data stream base on sliding window. In *Proceedings of 2nd International Conference on Future Computer and Communication (ICFCC)*, volume 2, pages V2–311. IEEE, 2010.

[103] G. Zhong and U. Hengartner. A distributed k-anonymity protocol for location privacy. In *Proceedings of International Conference on Pervasive Computing and Communications, PerCom*, pages 1–10. IEEE, 2009.